컴파일러 과제6 보고서 (2) - 테스트

20181755 이건희

2024년 11월

---

1. 개요

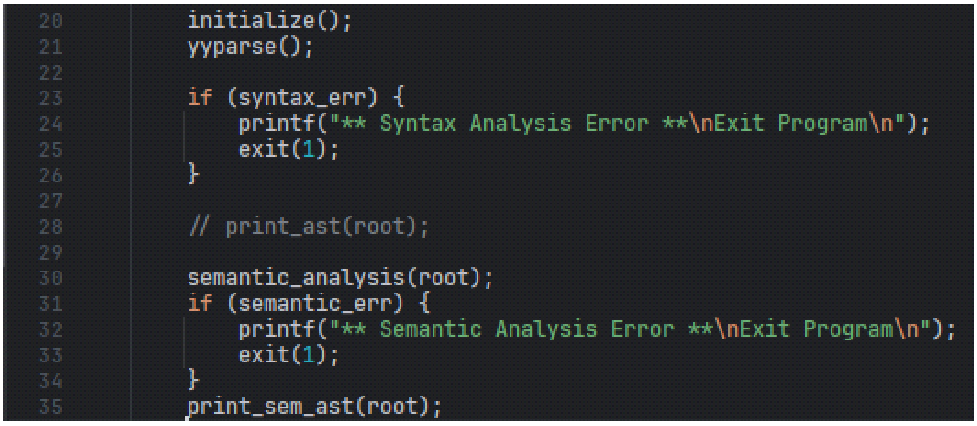
교수님께서 수업 중 변수의 선언과 동시에 할당을 하지 말라 하셔서, 이에 맞춰서 테스트 코드들을 완성했다. 테스트의 양이 많아 원본 코드와 리다이렉트한 원본 출력은 따로 첨부하고, 본 보고서에는 semantic tree만 집중적으로 관찰하기로 하기로 한다. 크게 오류가 없는 경우와 있는 경우로 나뉘는데, 각각의 경우 모두 gcc와 비교하였다.

전자는 4p부터, 후자는 41p부터 나옵니다.

2. print_ast와 semantic analysis의 관계

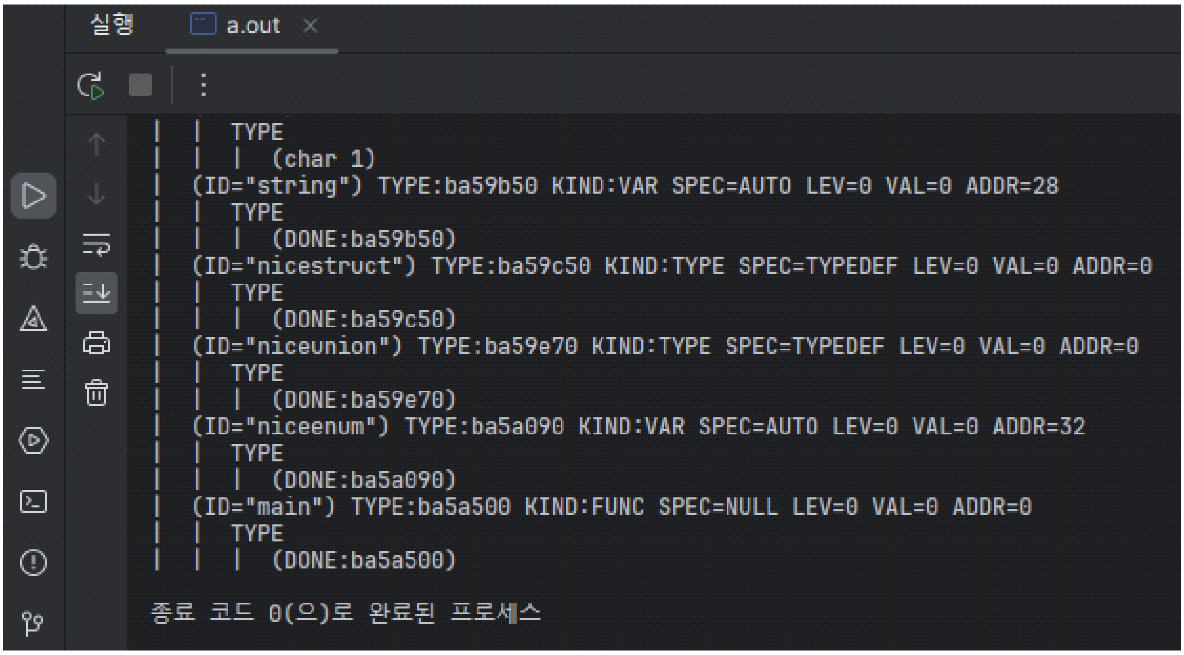
이 내용을 적어야 하나 고민을 했으나 일단은 적어본다.

본 과제는 semantic tree를 보고 분석하는 것을 목표라 생각하여, 메인 프로그램에서 syntax tree를 출력하는 print_ast를 주석처리 하고 프로그램을 실행해 봤는데 나와야 할 출력이 제대로 나오지 않았다.



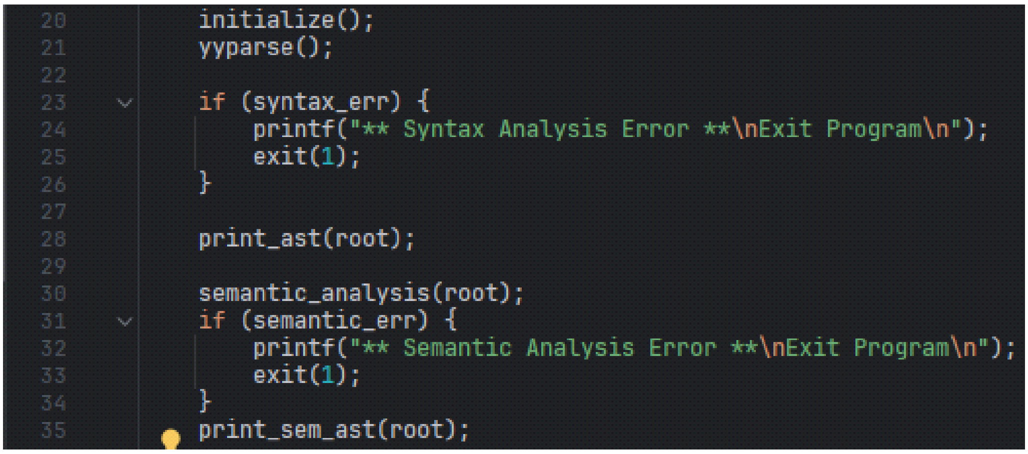
```
    initialize();
    yyparse();

    if (syntax_err) {
        printf("** Syntax Analysis Error **\nExit Program\n");
        exit(1);
    }

    // print_ast(root);

    semantic_analysis(root);
    if (semantic_err) {
        printf("** Semantic Analysis Error **\nExit Program\n");
        exit(1);
    }
    print_sem_ast(root);
```

print_ast()를 주석처리 하고 빌드하여 실행하면



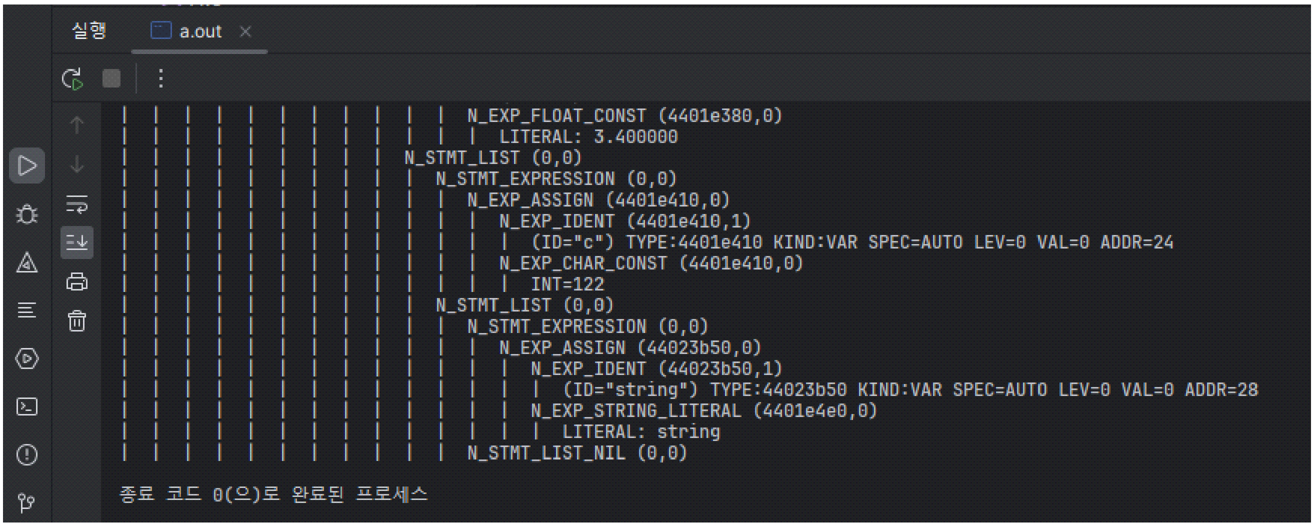
```
|  |  TYPE
|  |  |  (char 1)
|  (ID="string") TYPE:ba59b50 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=28
|  |  TYPE
|  |  |  (DONE:ba59b50)
|  (ID="nicestruct") TYPE:ba59c50 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  (DONE:ba59c50)
|  (ID="niceunion") TYPE:ba59e70 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  (DONE:ba59e70)
|  (ID="niceenum") TYPE:ba5a090 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=32
|  |  TYPE
|  |  |  (DONE:ba5a090)
|  (ID="main") TYPE:ba5a500 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  (DONE:ba5a500)

종료 코드 0(으)로 완료된 프로세스
```

할당과 관련된 내용을 전혀 출력하지 않는다.



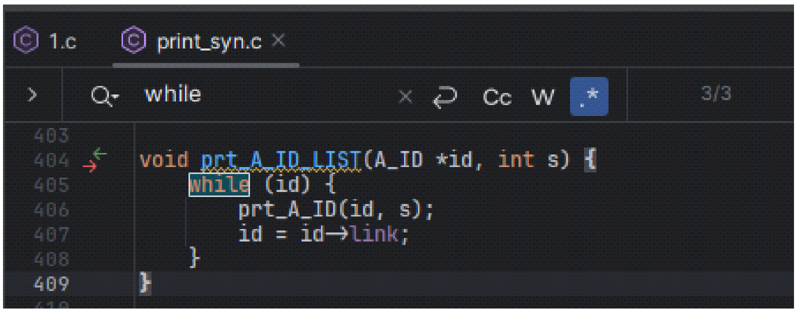
```c
    initialize();
    yyparse();

    if (syntax_err) {
        printf("** Syntax Analysis Error **\nExit Program\n");
        exit(1);
    }

    print_ast(root);

    semantic_analysis(root);
    if (semantic_err) {
        printf("** Semantic Analysis Error **\nExit Program\n");
        exit(1);
    }
    print_sem_ast(root);
```

print_ast()를 주석처리하지 않고 실행하면



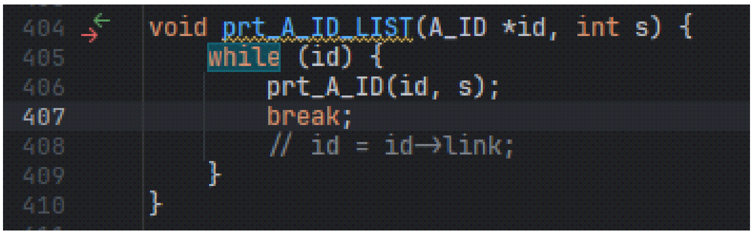
```
|   |   |   |   |   |   |   |   |   |   |   N_EXP_FLOAT_CONST (4401e380,0)
|   |   |   |   |   |   |   |   |   |   |   |   LITERAL: 3.400000
|   |   |   |   |   |   |   |   |   N_STMT_LIST (0,0)
|   |   |   |   |   |   |   |   |   |   N_STMT_EXPRESSION (0,0)
|   |   |   |   |   |   |   |   |   |   |   N_EXP_ASSIGN (4401e410,0)
|   |   |   |   |   |   |   |   |   |   |   |   N_EXP_IDENT (4401e410,1)
|   |   |   |   |   |   |   |   |   |   |   |   |   (ID="c") TYPE:4401e410 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=24
|   |   |   |   |   |   |   |   |   |   |   |   N_EXP_CHAR_CONST (4401e410,0)
|   |   |   |   |   |   |   |   |   |   |   |   |   INT=122
|   |   |   |   |   |   |   |   |   |   N_STMT_LIST (0,0)
|   |   |   |   |   |   |   |   |   |   |   N_STMT_EXPRESSION (0,0)
|   |   |   |   |   |   |   |   |   |   |   |   N_EXP_ASSIGN (44023b50,0)
|   |   |   |   |   |   |   |   |   |   |   |   |   N_EXP_IDENT (44023b50,1)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   (ID="string") TYPE:44023b50 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=28
|   |   |   |   |   |   |   |   |   |   |   |   |   N_EXP_STRING_LITERAL (4401e4e0,0)
|   |   |   |   |   |   |   |   |   |   |   |   |   |   LITERAL: string
|   |   |   |   |   |   |   |   |   |   |   N_STMT_LIST_NIL (0,0)

종료 코드 0(으)로 완료된 프로세스
```

정상적으로 출력이 된다.



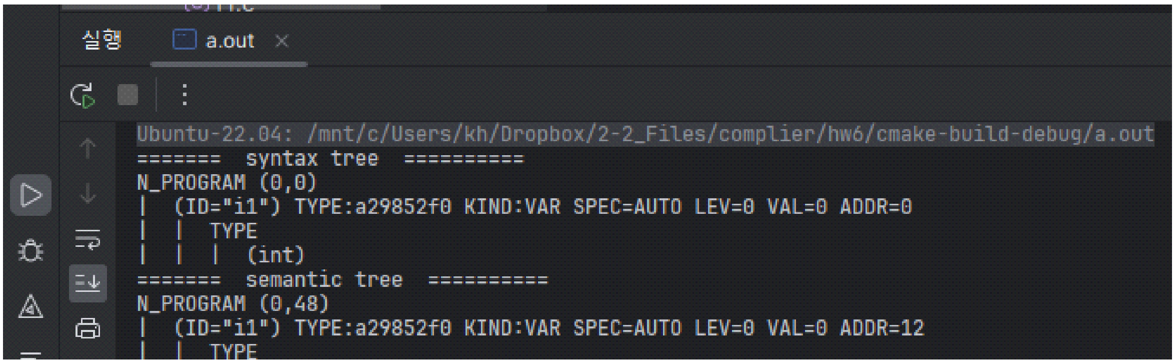
```c
void prt_A_ID_LIST(A_ID *id, int s) {
    while (id) {
        prt_A_ID(id, s);
        id = id→link;
    }
}
```

이의 이유가 궁금하여 코드를 살펴보니 print_syn.c의 prt_A_ID_LIST()가 원인으로 생각되었다. 해당 함수는 while loop을 돌며 syntax table을 출력하며, A_ID id를 link에 있는 다음 A_ID로 바꾼다.



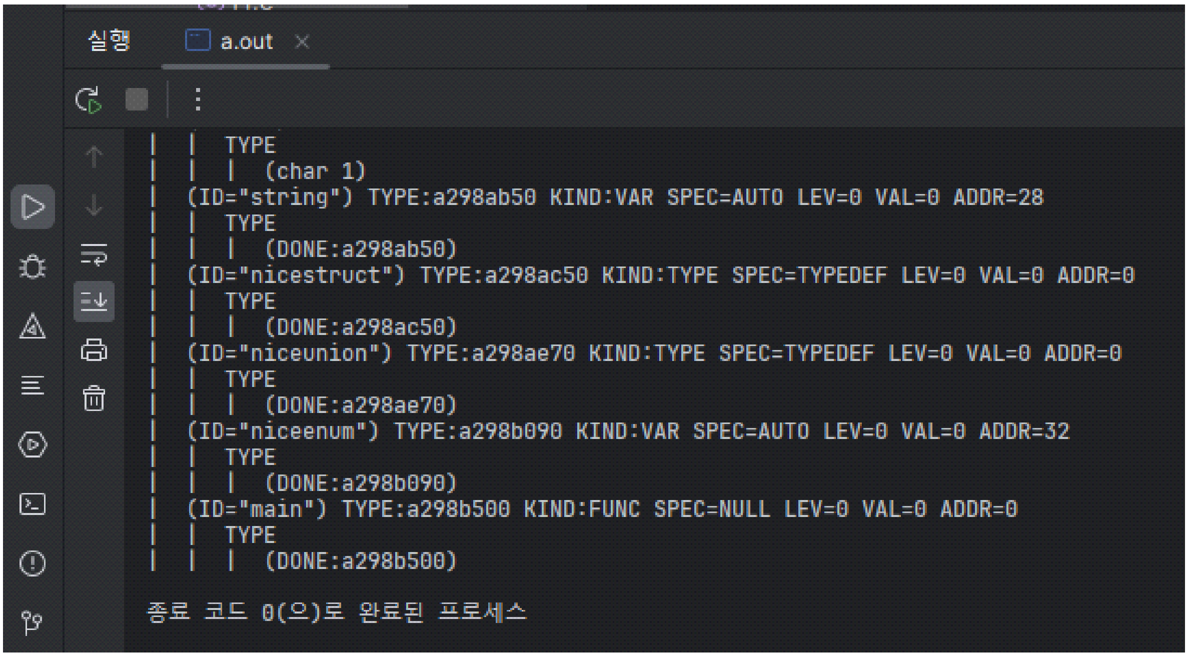
```
void prt_A_ID_LIST(A_ID *id, int s) {
    while (id) {
        prt_A_ID(id, s);
        break;
        // id = id->link;
    }
}
```

이를 수행하지 않게끔 수정하고 실행해보면



```
Ubuntu-22.04: /mnt/c/Users/kh/Dropbox/2-2_Files/complier/hw6/cmake-build-debug/a.out
=======  syntax tree  ==========
N_PROGRAM (0,0)
|  (ID="i1") TYPE:a29852f0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  (int)
=======  semantic tree  ==========
N_PROGRAM (0,48)
|  (ID="i1") TYPE:a29852f0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=12
|  |  TYPE
```

syntax tree가 당연하게도 정상적으로 출력되지 않는다(이것은 이해가 됨). 노드 N_PROGRAM에 달린 모든 syntax table을 출력하지 못하고, 첫 syntax table만 출력하고 끝나는 것이었다.



```
|  |  TYPE
|  |  |  (char 1)
|  (ID="string") TYPE:a298ab50 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=28
|  |  TYPE
|  |  |  (DONE:a298ab50)
|  (ID="nicestruct") TYPE:a298ac50 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  (DONE:a298ac50)
|  (ID="niceunion") TYPE:a298ae70 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  (DONE:a298ae70)
|  (ID="niceenum") TYPE:a298b090 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=32
|  |  TYPE
|  |  |  (DONE:a298b090)
|  (ID="main") TYPE:a298b500 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  (DONE:a298b500)

종료 코드 0(으)로 완료된 프로세스
```

semantic tree는 print_ast() 전체를 주석처리 한 것과 똑같이 나왔다(이것은 이해가 되지 않음).

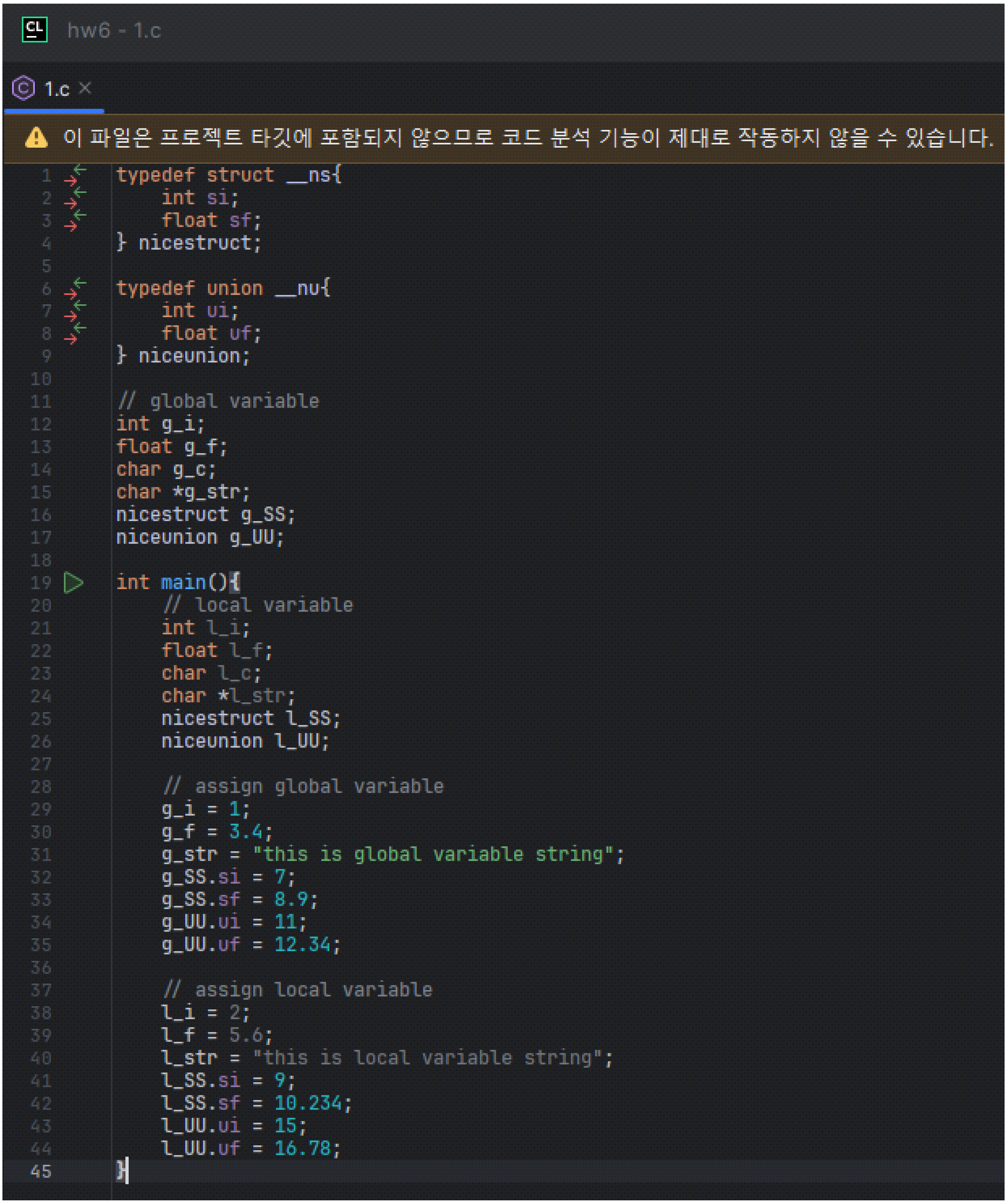
메인 프로그램의 흐름 상 print_ast() 다음에는 semantic_analysis() -> sem_program()으로부터 노드 N_PROGRAM부터 시작하여 탑다운으로 타고 내려가는데, print_ast()에서 syntax table을 전부 출력하지 않은 것이 왜 semantic analysis까지 영향을 미치는지는 솔직히 모르겠다.

보고서에는 syntax tree의 출력을 생략하도록 하겠다.

3. 테스트 (1) - 올바른 코드

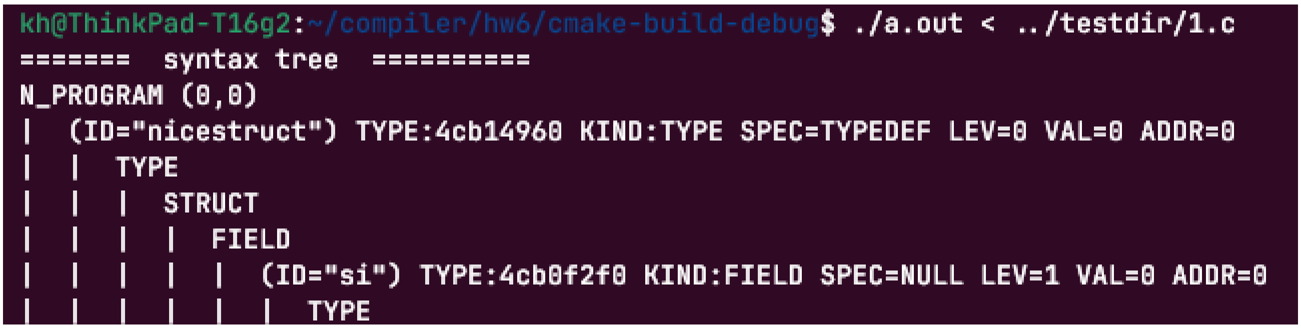
파일 하나에 여러가지 수식, 선언을 넣었습니다.

3-1. 1.c - global variable/local variable declaration, assign



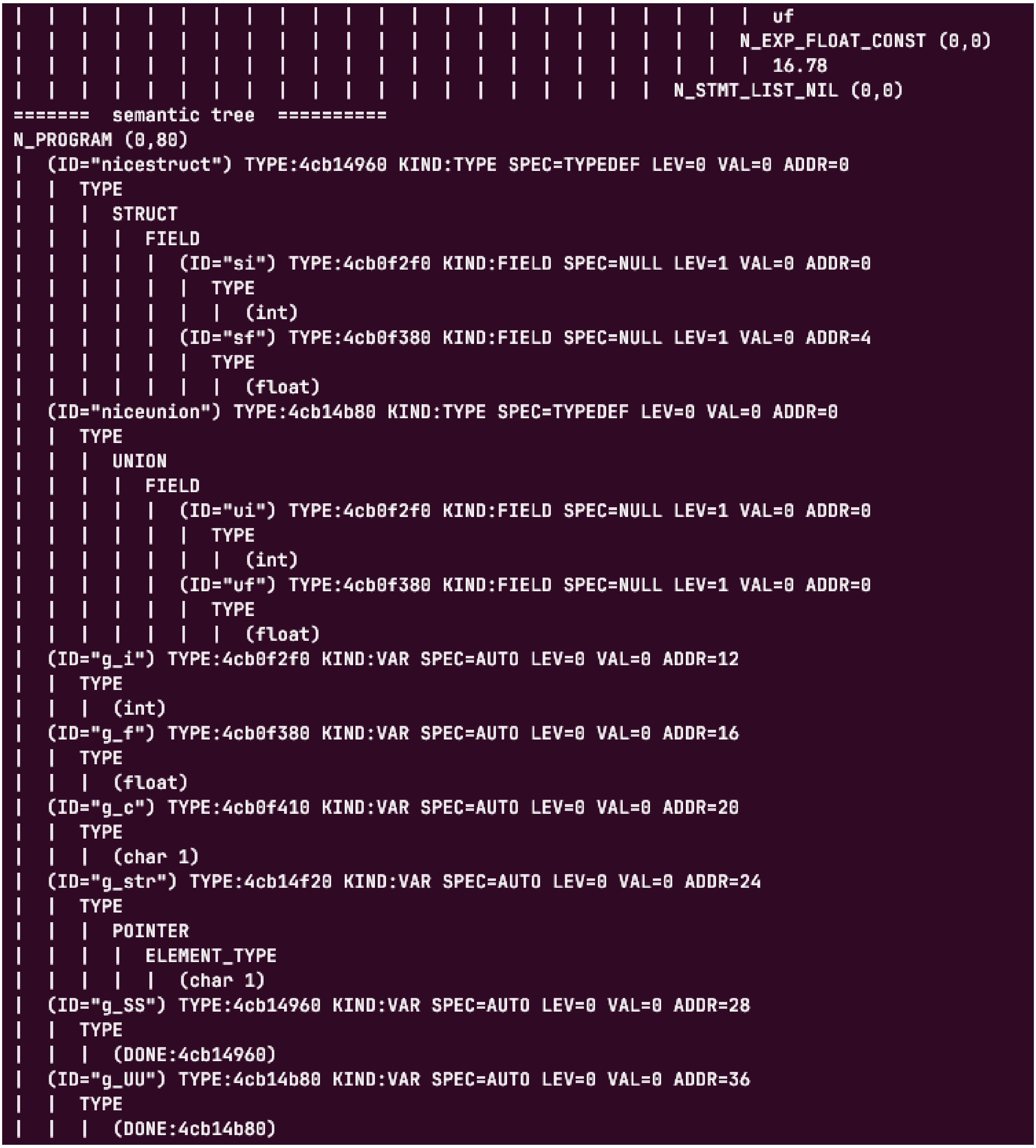
이 파일은 프로젝트 타깃에 포함되지 않으므로 코드 분석 기능이 제대로 작동하지 않을 수 있습니다.

```c
typedef struct __ns{
    int si;
    float sf;
} nicestruct;

typedef union __nu{
    int ui;
    float uf;
} niceunion;

// global variable
int g_i;
float g_f;
char g_c;
char *g_str;
nicestruct g_SS;
niceunion g_UU;

int main(){
    // local variable
    int l_i;
    float l_f;
    char l_c;
    char *l_str;
    nicestruct l_SS;
    niceunion l_UU;

    // assign global variable
    g_i = 1;
    g_f = 3.4;
    g_str = "this is global variable string";
    g_SS.si = 7;
    g_SS.sf = 8.9;
    g_UU.ui = 11;
    g_UU.uf = 12.34;

    // assign local variable
    l_i = 2;
    l_f = 5.6;
    l_str = "this is local variable string";
    l_SS.si = 9;
    l_SS.sf = 10.234;
    l_UU.ui = 15;
    l_UU.uf = 16.78;
}
```

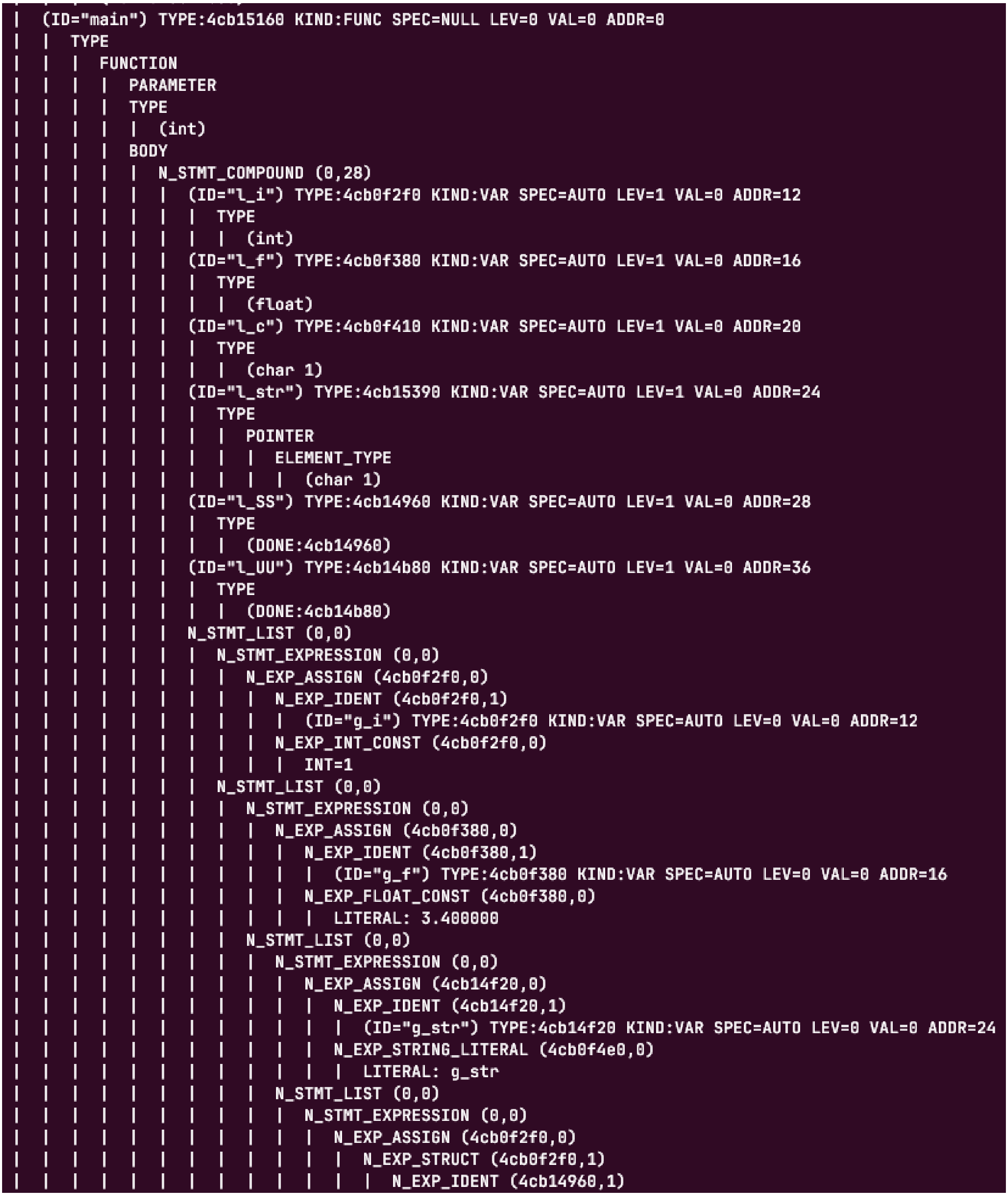


```
kh@ThinkPad-T16g2:~/compiler/hw6/cmake-build-debug$ ./a.out < ../testdir/1.c
=======  syntax tree  ==========
N_PROGRAM (0,0)
|  (ID="nicestruct") TYPE:4cb14960 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  STRUCT
|  |  |  |  FIELD
|  |  |  |  |  (ID="si") TYPE:4cb0f2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  TYPE
```

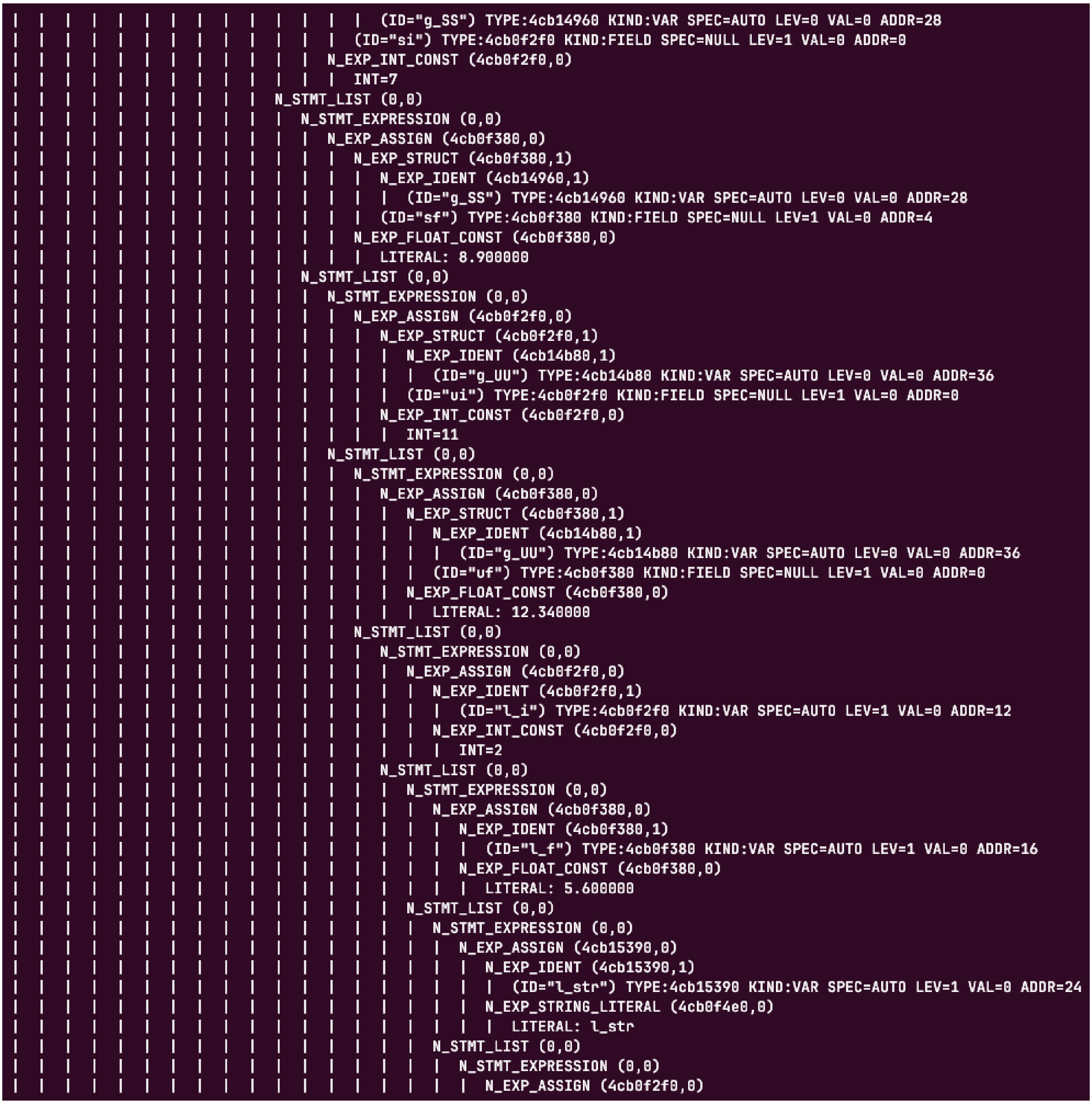
(syntax tree 생략)



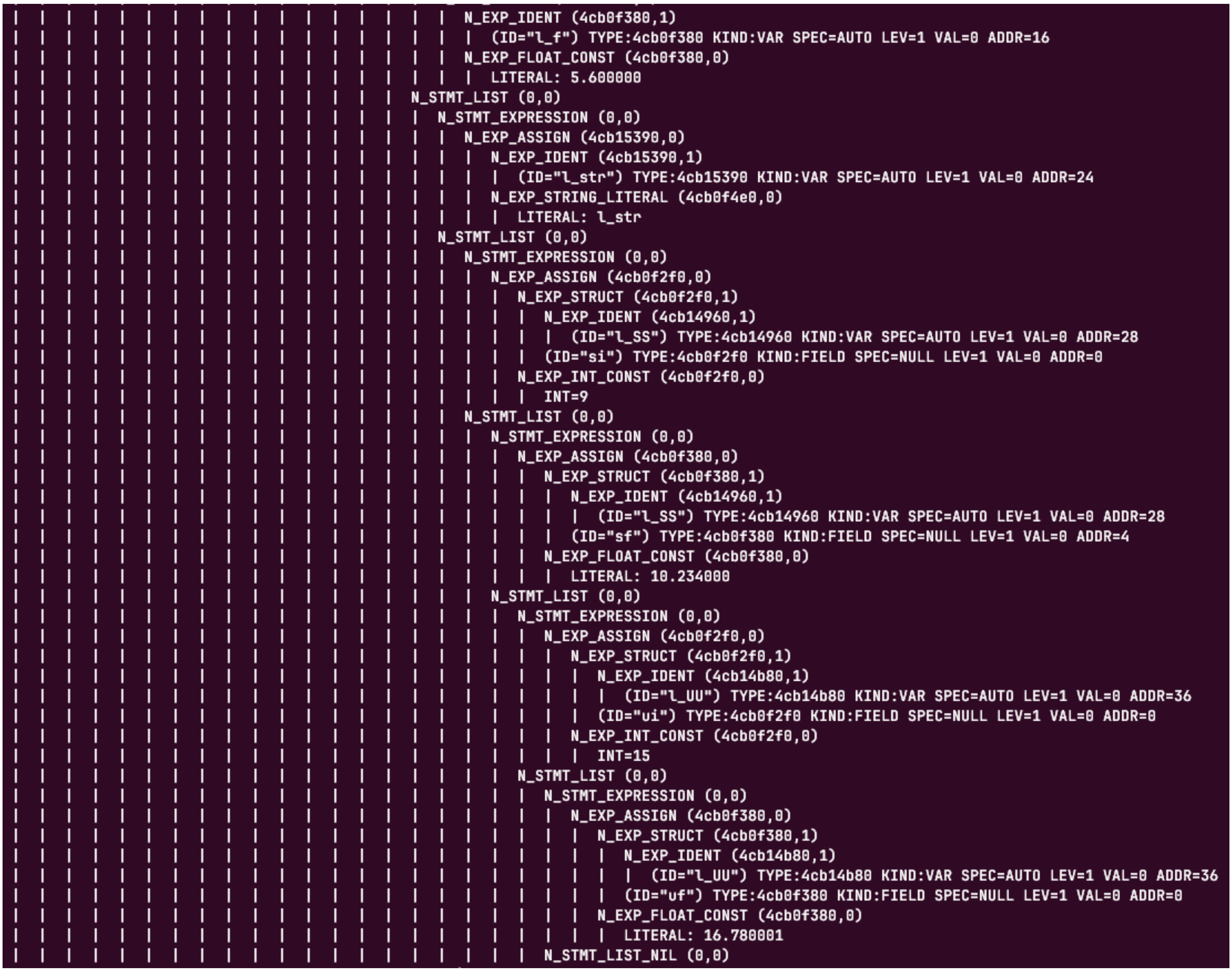
```
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  uf
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FLOAT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  16.78
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
=======  semantic tree  ==========
N_PROGRAM (0,80)
|  (ID="nicestruct") TYPE:4cb14960 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  STRUCT
|  |  |  |  FIELD
|  |  |  |  |  (ID="si") TYPE:4cb0f2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (int)
|  |  |  |  |  (ID="sf") TYPE:4cb0f380 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=4
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (float)
|  (ID="niceunion") TYPE:4cb14b80 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  UNION
|  |  |  |  FIELD
|  |  |  |  |  (ID="ui") TYPE:4cb0f2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (int)
|  |  |  |  |  (ID="uf") TYPE:4cb0f380 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (float)
|  (ID="g_i") TYPE:4cb0f2f0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=12
|  |  TYPE
|  |  |  (int)
|  (ID="g_f") TYPE:4cb0f380 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=16
|  |  TYPE
|  |  |  (float)
|  (ID="g_c") TYPE:4cb0f410 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=20
|  |  TYPE
|  |  |  (char 1)
|  (ID="g_str") TYPE:4cb14f20 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=24
|  |  TYPE
|  |  |  POINTER
|  |  |  |  ELEMENT_TYPE
|  |  |  |  |  (char 1)
|  (ID="g_SS") TYPE:4cb14960 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=28
|  |  TYPE
|  |  |  (DONE:4cb14960)
|  (ID="g_UU") TYPE:4cb14b80 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=36
|  |  TYPE
|  |  |  (DONE:4cb14b80)
```



```
|  (ID="main") TYPE:4cb15160 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
|  |  |  |  |  (int)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,28)
|  |  |  |  |  |  (ID="l_i") TYPE:4cb0f2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="l_f") TYPE:4cb0f380 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (float)
|  |  |  |  |  |  (ID="l_c") TYPE:4cb0f410 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (char 1)
|  |  |  |  |  |  (ID="l_str") TYPE:4cb15390 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  POINTER
|  |  |  |  |  |  |  |  |  ELEMENT_TYPE
|  |  |  |  |  |  |  |  |  |  (char 1)
|  |  |  |  |  |  (ID="l_SS") TYPE:4cb14960 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=28
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (DONE:4cb14960)
|  |  |  |  |  |  (ID="l_UU") TYPE:4cb14b80 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=36
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (DONE:4cb14b80)
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb0f2f0,1)
|  |  |  |  |  |  |  |  |  |  (ID="g_i") TYPE:4cb0f2f0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb0f380,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="g_f") TYPE:4cb0f380 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  N_EXP_FLOAT_CONST (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  LITERAL: 3.400000
|  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb14f20,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb14f20,1)
|  |  |  |  |  |  |  |  |  |  |  |  (ID="g_str") TYPE:4cb14f20 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=24
|  |  |  |  |  |  |  |  |  |  |  N_EXP_STRING_LITERAL (4cb0f4e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  LITERAL: g_str
|  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRUCT (4cb0f2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb14960,1)
```

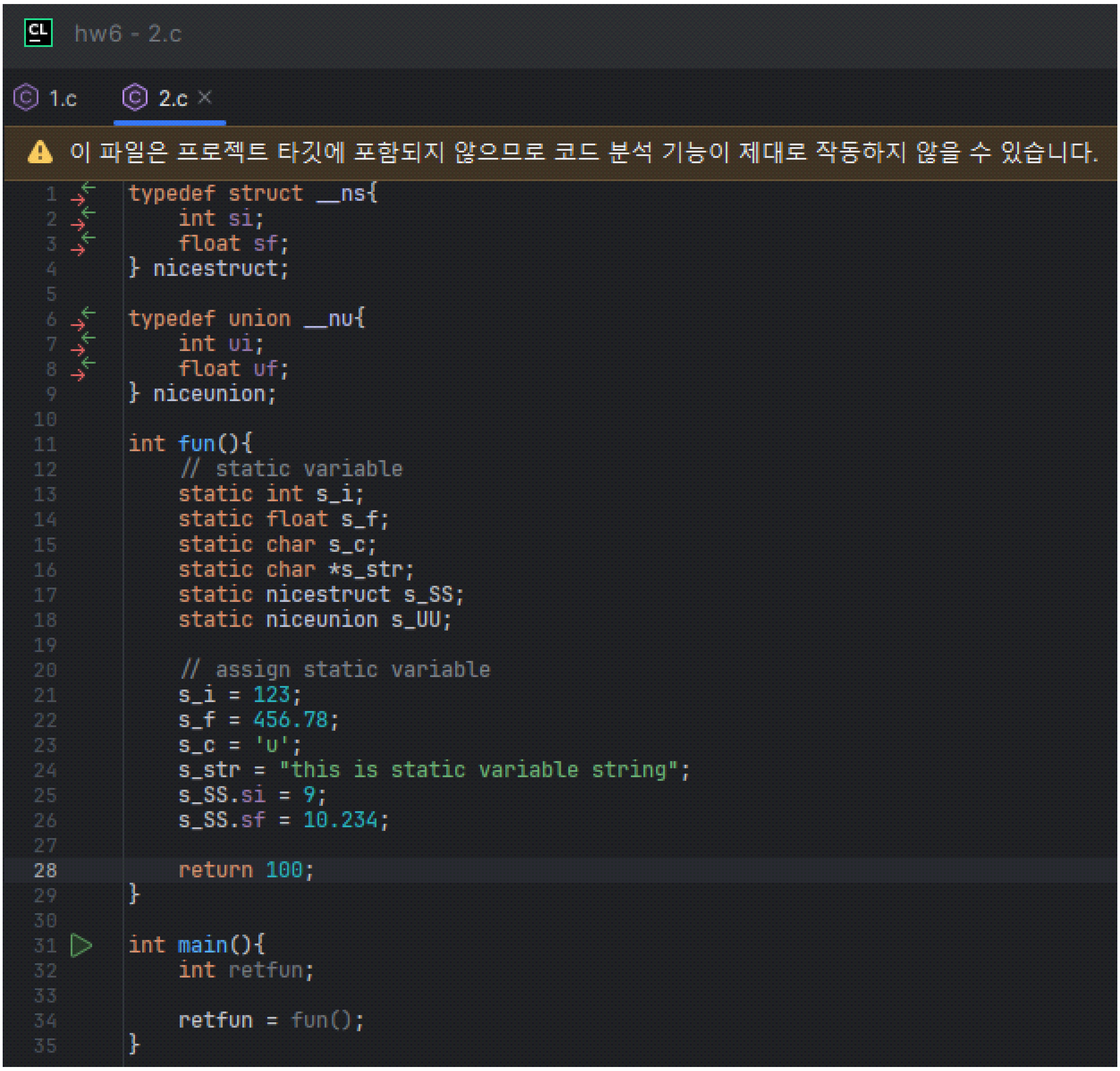


```
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="g_SS") TYPE:4cb14960 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=28
|  |  |  |  |  |  |  |  |  |  |  |  |  (ID="si") TYPE:4cb0f2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  INT=7
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRUCT (4cb0f380,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb14960,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="g_SS") TYPE:4cb14960 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=28
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="sf") TYPE:4cb0f380 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=4
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FLOAT_CONST (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: 8.900000
|  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRUCT (4cb0f2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb14b80,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="g_UU") TYPE:4cb14b80 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=36
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="ui") TYPE:4cb0f2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=11
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRUCT (4cb0f380,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb14b80,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="g_UU") TYPE:4cb14b80 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=36
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="uf") TYPE:4cb0f380 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FLOAT_CONST (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: 12.340000
|  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb0f2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="l_i") TYPE:4cb0f2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=2
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb0f380,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="l_f") TYPE:4cb0f380 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FLOAT_CONST (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: 5.600000
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb15390,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb15390,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="l_str") TYPE:4cb15390 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRING_LITERAL (4cb0f4e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: l_str
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f2f0,0)
```

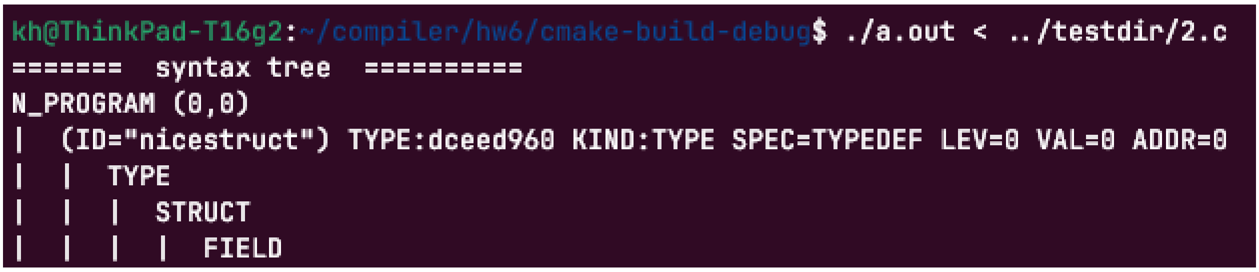


```
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb0f380,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="l_f") TYPE:4cb0f380 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FLOAT_CONST (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: 5.600000
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb15390,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb15390,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="l_str") TYPE:4cb15390 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRING_LITERAL (4cb0f4e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: l_str
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRUCT (4cb0f2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb14960,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="l_SS") TYPE:4cb14960 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=28
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="si") TYPE:4cb0f2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=9
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRUCT (4cb0f380,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb14960,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="l_SS") TYPE:4cb14960 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=28
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="sf") TYPE:4cb0f380 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=4
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FLOAT_CONST (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: 10.234000
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRUCT (4cb0f2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb14b80,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="l_UU") TYPE:4cb14b80 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=36
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="ui") TYPE:4cb0f2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (4cb0f2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=15
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRUCT (4cb0f380,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (4cb14b80,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="l_UU") TYPE:4cb14b80 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=36
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="uf") TYPE:4cb0f380 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FLOAT_CONST (4cb0f380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: 16.780001
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
```

3-2. 2.c - static variable declaration, assign

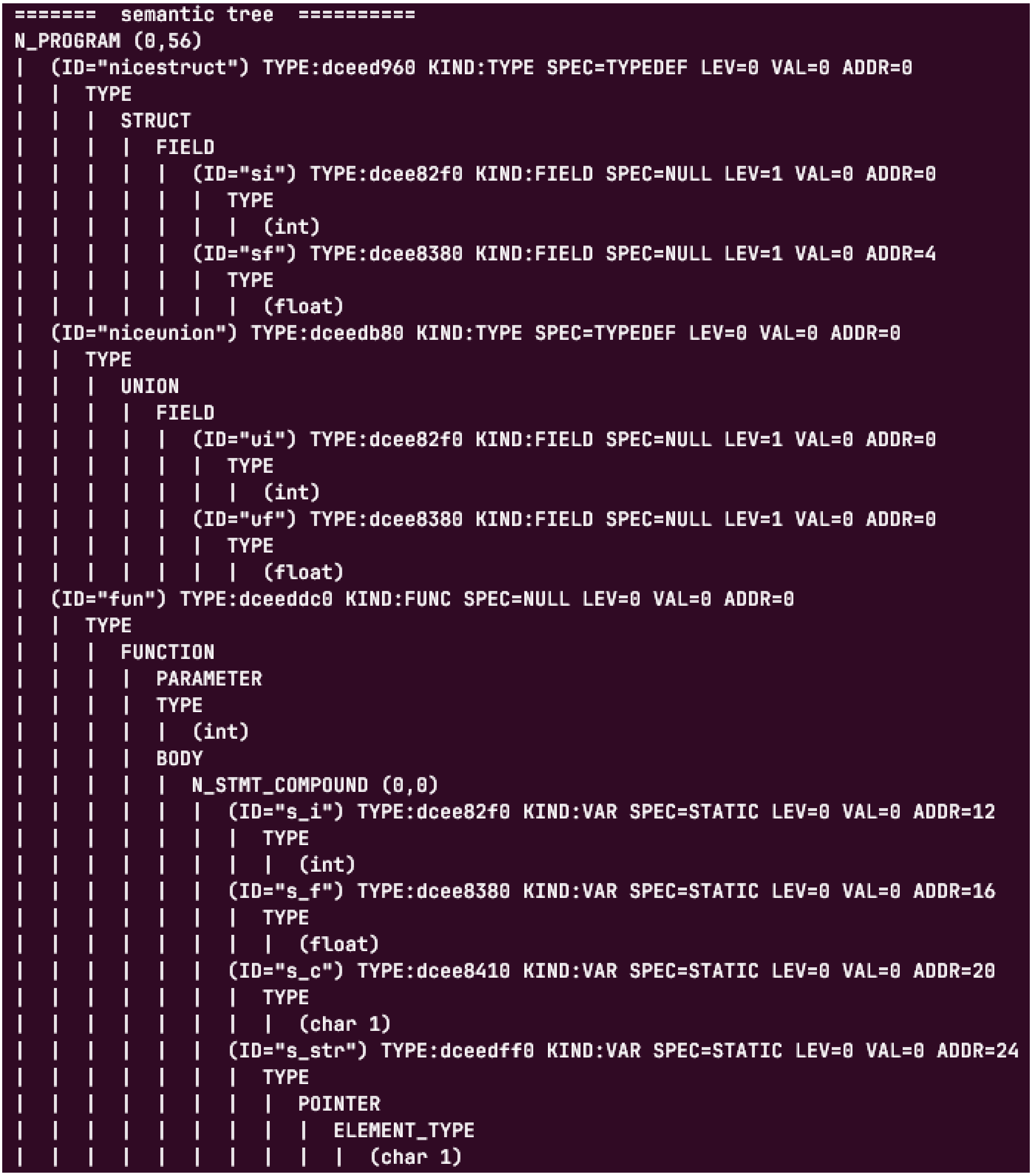

hw6 - 2.c
1.c
2.c ×
이 파일은 프로젝트 타깃에 포함되지 않으므로 코드 분석 기능이 제대로 작동하지 않을 수 있습니다.

```c
typedef struct __ns{
    int si;
    float sf;
} nicestruct;

typedef union __nu{
    int ui;
    float uf;
} niceunion;

int fun(){
    // static variable
    static int s_i;
    static float s_f;
    static char s_c;
    static char *s_str;
    static nicestruct s_SS;
    static niceunion s_UU;

    // assign static variable
    s_i = 123;
    s_f = 456.78;
    s_c = 'u';
    s_str = "this is static variable string";
    s_SS.si = 9;
    s_SS.sf = 10.234;

    return 100;
}

int main(){
    int retfun;

    retfun = fun();
}
```

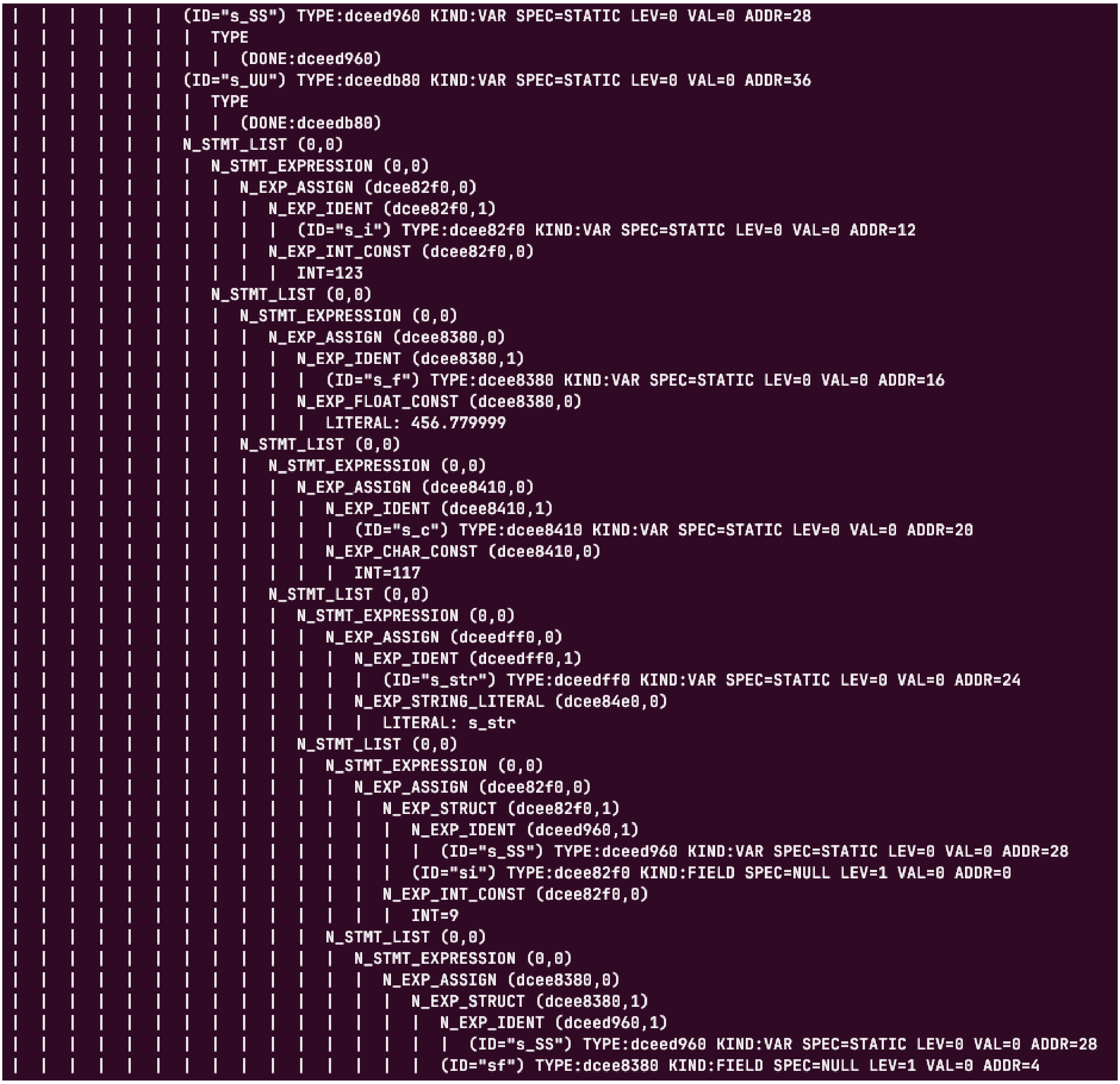


```
kh@ThinkPad-T16g2:~/compiler/hw6/cmake-build-debug$ ./a.out < ../testdir/2.c
=======  syntax tree  ==========
N_PROGRAM (0,0)
|  (ID="nicestruct") TYPE:dceed960 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  STRUCT
|  |  |  |  FIELD
```

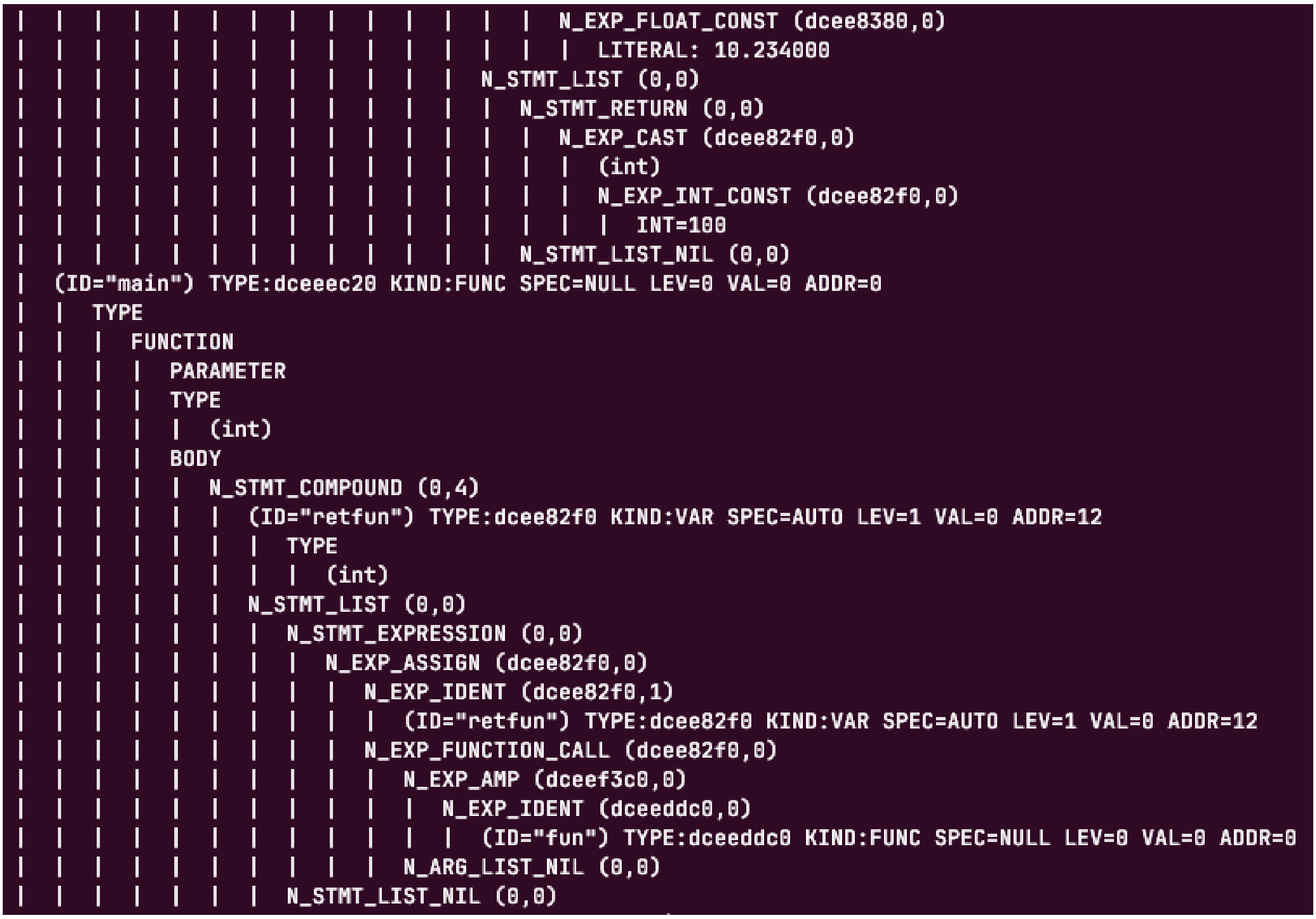
(syntax tree 생략)



```
=======  semantic tree  ==========
N_PROGRAM (0,56)
|  (ID="nicestruct") TYPE:dceed960 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  STRUCT
|  |  |  |  FIELD
|  |  |  |  |  (ID="si") TYPE:dcee82f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (int)
|  |  |  |  |  (ID="sf") TYPE:dcee8380 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=4
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (float)
|  (ID="niceunion") TYPE:dceedb80 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  UNION
|  |  |  |  FIELD
|  |  |  |  |  (ID="ui") TYPE:dcee82f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (int)
|  |  |  |  |  (ID="uf") TYPE:dcee8380 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (float)
|  (ID="fun") TYPE:dceeddc0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
|  |  |  |  |  (int)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  (ID="s_i") TYPE:dcee82f0 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=12
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="s_f") TYPE:dcee8380 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=16
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (float)
|  |  |  |  |  |  (ID="s_c") TYPE:dcee8410 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=20
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (char 1)
|  |  |  |  |  |  (ID="s_str") TYPE:dceedff0 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=24
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  POINTER
|  |  |  |  |  |  |  |  |  ELEMENT_TYPE
|  |  |  |  |  |  |  |  |  |  (char 1)
```

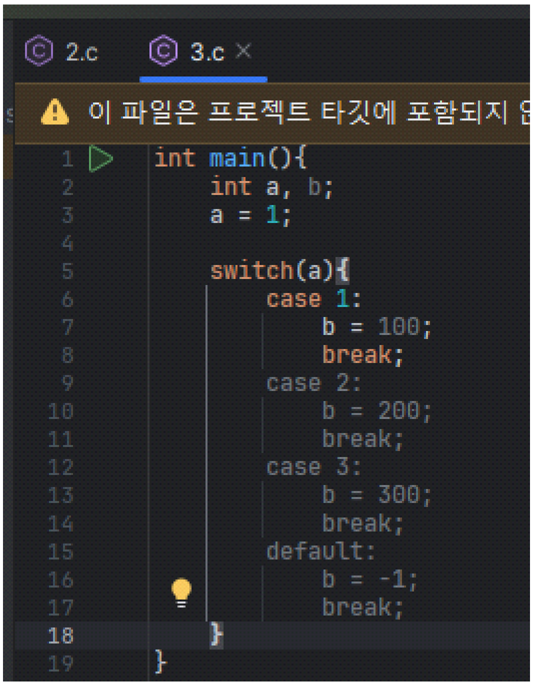


```
|  |  |  |  |  |  (ID="s_SS") TYPE:dceed960 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=28
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (DONE:dceed960)
|  |  |  |  |  |  (ID="s_UU") TYPE:dceedb80 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=36
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (DONE:dceedb80)
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  N_EXP_ASSIGN (dcee82f0,0)
|  |  |  |  |  |  |  |  |  N_EXP_IDENT (dcee82f0,1)
|  |  |  |  |  |  |  |  |  |  (ID="s_i") TYPE:dcee82f0 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (dcee82f0,0)
|  |  |  |  |  |  |  |  |  |  INT=123
|  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (dcee8380,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (dcee8380,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="s_f") TYPE:dcee8380 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  N_EXP_FLOAT_CONST (dcee8380,0)
|  |  |  |  |  |  |  |  |  |  |  LITERAL: 456.779999
|  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (dcee8410,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (dcee8410,1)
|  |  |  |  |  |  |  |  |  |  |  |  (ID="s_c") TYPE:dcee8410 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  |  N_EXP_CHAR_CONST (dcee8410,0)
|  |  |  |  |  |  |  |  |  |  |  |  INT=117
|  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (dceedff0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (dceedff0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  (ID="s_str") TYPE:dceedff0 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=24
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRING_LITERAL (dcee84e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: s_str
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (dcee82f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRUCT (dcee82f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (dceed960,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="s_SS") TYPE:dceed960 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=28
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="si") TYPE:dcee82f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (dcee82f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=9
|  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (dcee8380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRUCT (dcee8380,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (dceed960,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="s_SS") TYPE:dceed960 KIND:VAR SPEC=STATIC LEV=0 VAL=0 ADDR=28
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="sf") TYPE:dcee8380 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=4
```



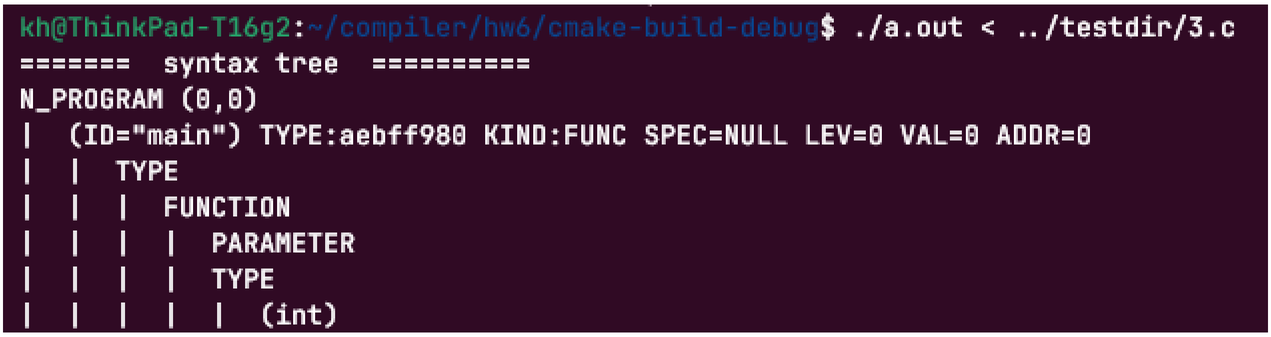
```
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FLOAT_CONST (dcee8380,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: 10.234000
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_RETURN (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (dcee82f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (dcee82f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=100
|  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  (ID="main") TYPE:dceeec20 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
|  |  |  |  |  (int)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,4)
|  |  |  |  |  |  (ID="retfun") TYPE:dcee82f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  N_EXP_ASSIGN (dcee82f0,0)
|  |  |  |  |  |  |  |  |  N_EXP_IDENT (dcee82f0,1)
|  |  |  |  |  |  |  |  |  |  (ID="retfun") TYPE:dcee82f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (dcee82f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_AMP (dceef3c0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (dceeddc0,0)
|  |  |  |  |  |  |  |  |  |  |  |  (ID="fun") TYPE:dceeddc0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
```

3-3. 3.c - compound statement : switch, labled statement



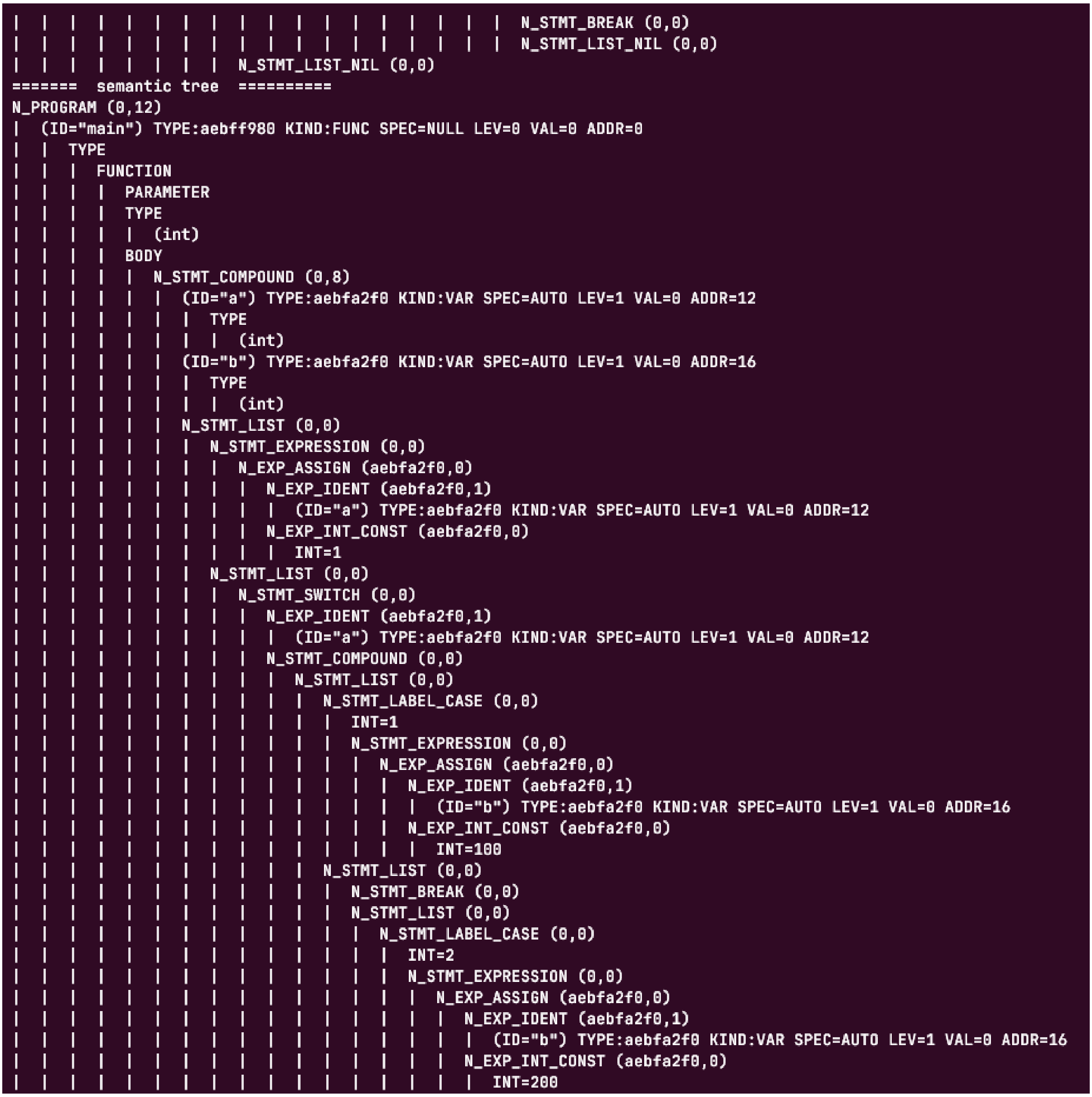
⚠ 이 파일은 프로젝트 타깃에 포함되지 않

```c
int main(){
    int a, b;
    a = 1;

    switch(a){
        case 1:
            b = 100;
            break;
        case 2:
            b = 200;
            break;
        case 3:
            b = 300;
            break;
        default:
            b = -1;
            break;
    }
}
```

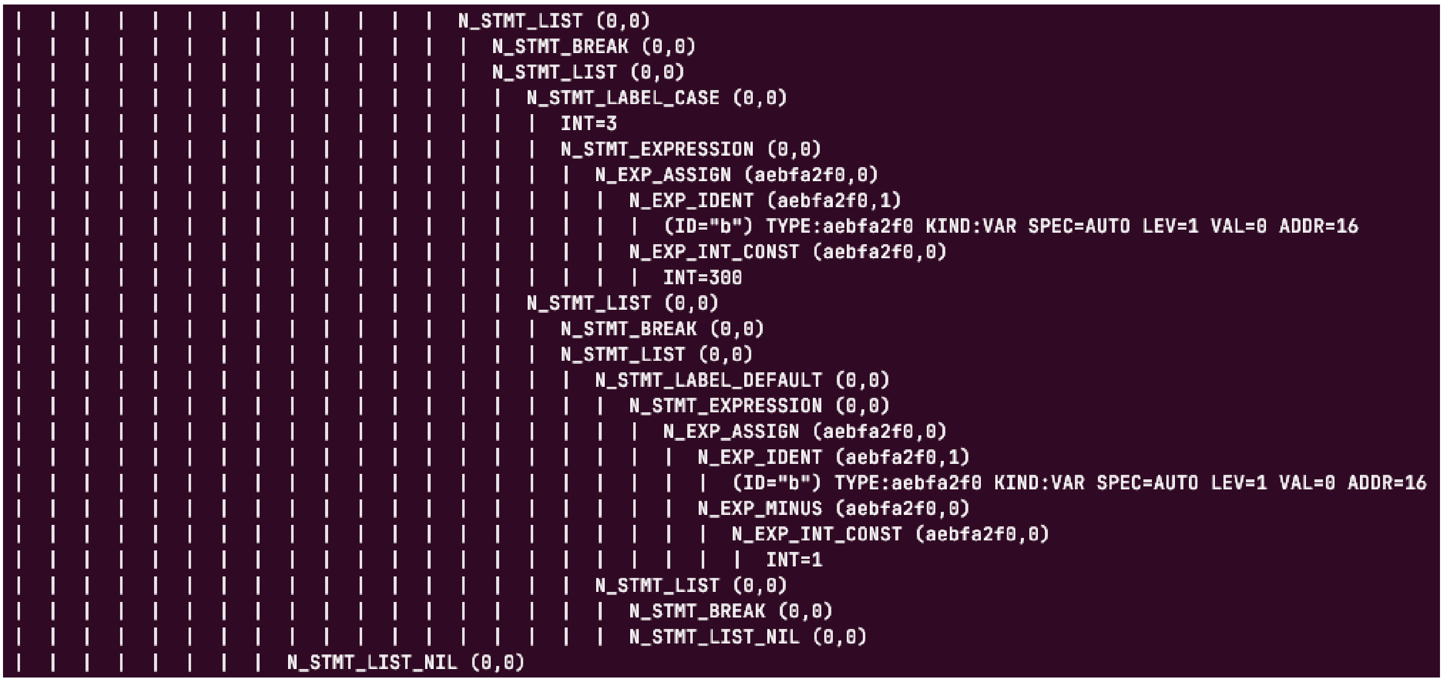


```
kh@ThinkPad-T16g2:~/compiler/hw6/cmake-build-debug$ ./a.out < ../testdir/3.c
=======  syntax tree  ==========
N_PROGRAM (0,0)
|  (ID="main") TYPE:aebff980 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
|  |  |  |  |  (int)
```

(syntax tree 생략)

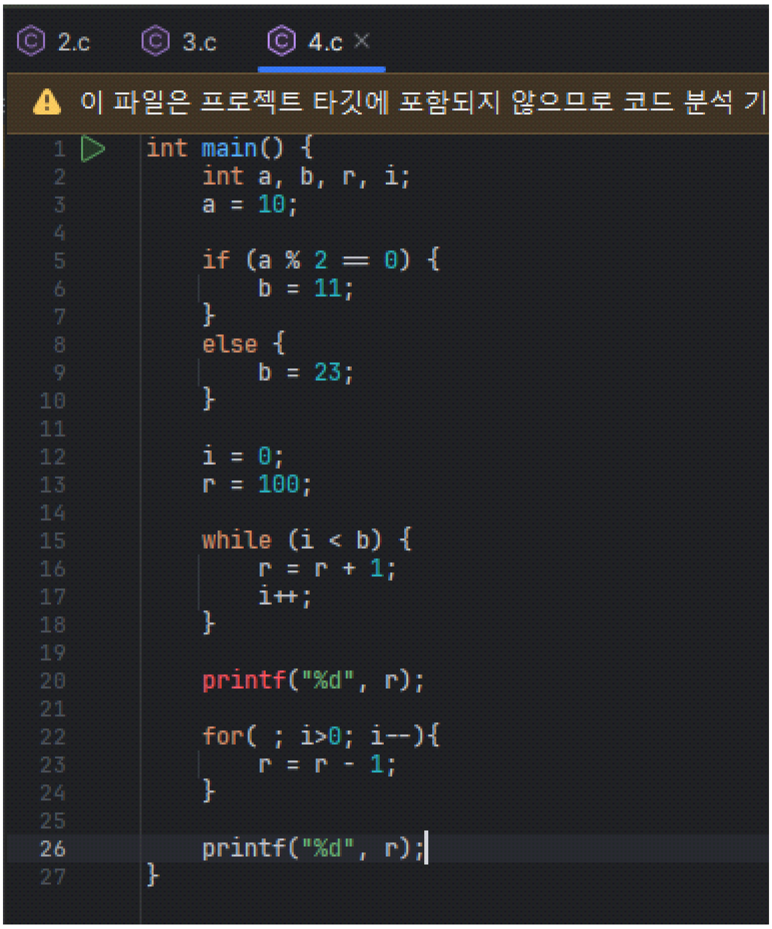


```
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_BREAK (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
=======  semantic tree  ==========
N_PROGRAM (0,12)
|  (ID="main") TYPE:aebff980 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
|  |  |  |  |  (int)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,8)
|  |  |  |  |  |  (ID="a") TYPE:aebfa2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="b") TYPE:aebfa2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  N_EXP_ASSIGN (aebfa2f0,0)
|  |  |  |  |  |  |  |  |  N_EXP_IDENT (aebfa2f0,1)
|  |  |  |  |  |  |  |  |  |  (ID="a") TYPE:aebfa2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (aebfa2f0,0)
|  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  N_STMT_SWITCH (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_IDENT (aebfa2f0,1)
|  |  |  |  |  |  |  |  |  |  (ID="a") TYPE:aebfa2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_STMT_LABEL_CASE (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (aebfa2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (aebfa2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:aebfa2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (aebfa2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=100
|  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_BREAK (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LABEL_CASE (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=2
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (aebfa2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (aebfa2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:aebfa2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (aebfa2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=200
```



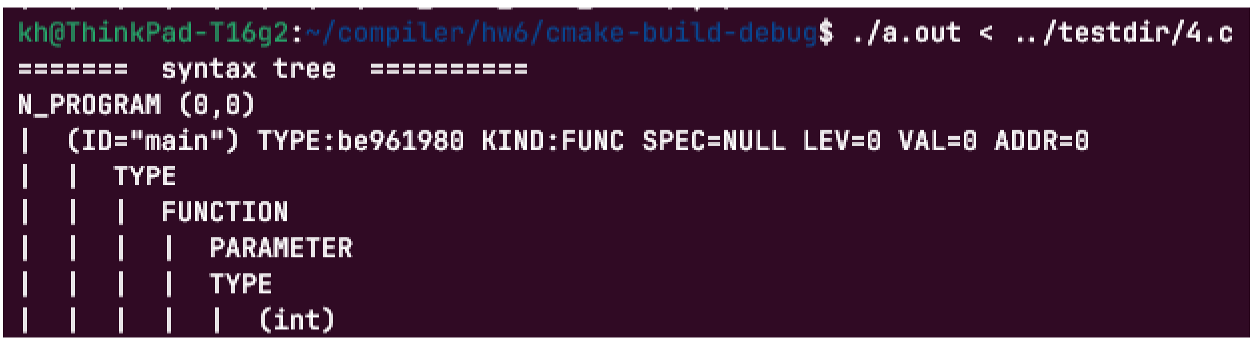
```
|  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_BREAK (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LABEL_CASE (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=3
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (aebfa2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (aebfa2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:aebfa2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (aebfa2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=300
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_BREAK (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LABEL_DEFAULT (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (aebfa2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (aebfa2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:aebfa2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_MINUS (aebfa2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (aebfa2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_BREAK (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
```

3-4. 4.c - compound statement : if~else, while, for



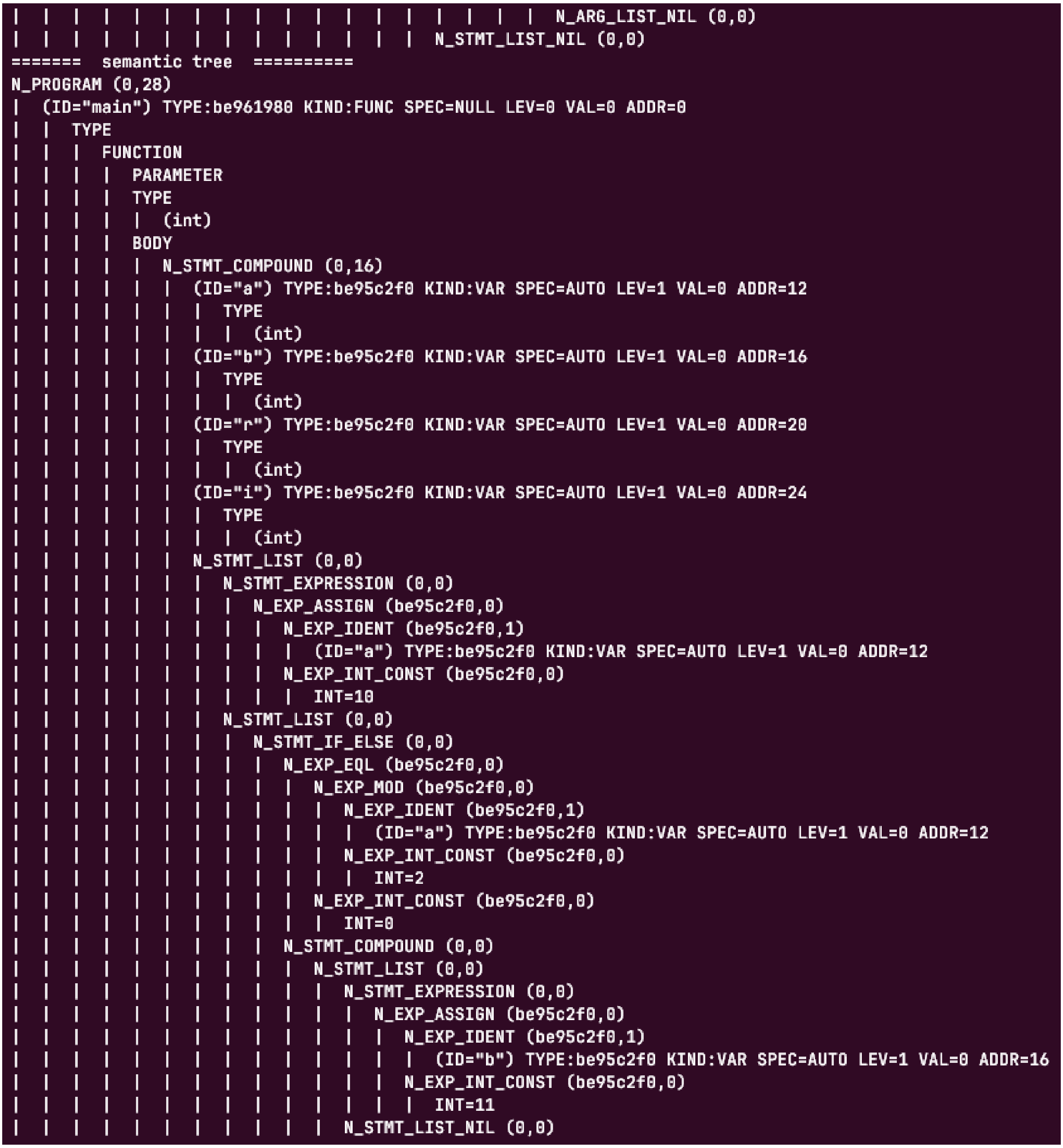
이 파일은 프로젝트 타깃에 포함되지 않으므로 코드 분석 기

```c
int main() {
    int a, b, r, i;
    a = 10;

    if (a % 2 == 0) {
        b = 11;
    }
    else {
        b = 23;
    }

    i = 0;
    r = 100;

    while (i < b) {
        r = r + 1;
        i++;
    }

    printf("%d", r);

    for( ; i>0; i--){
        r = r - 1;
    }

    printf("%d", r);
}
```

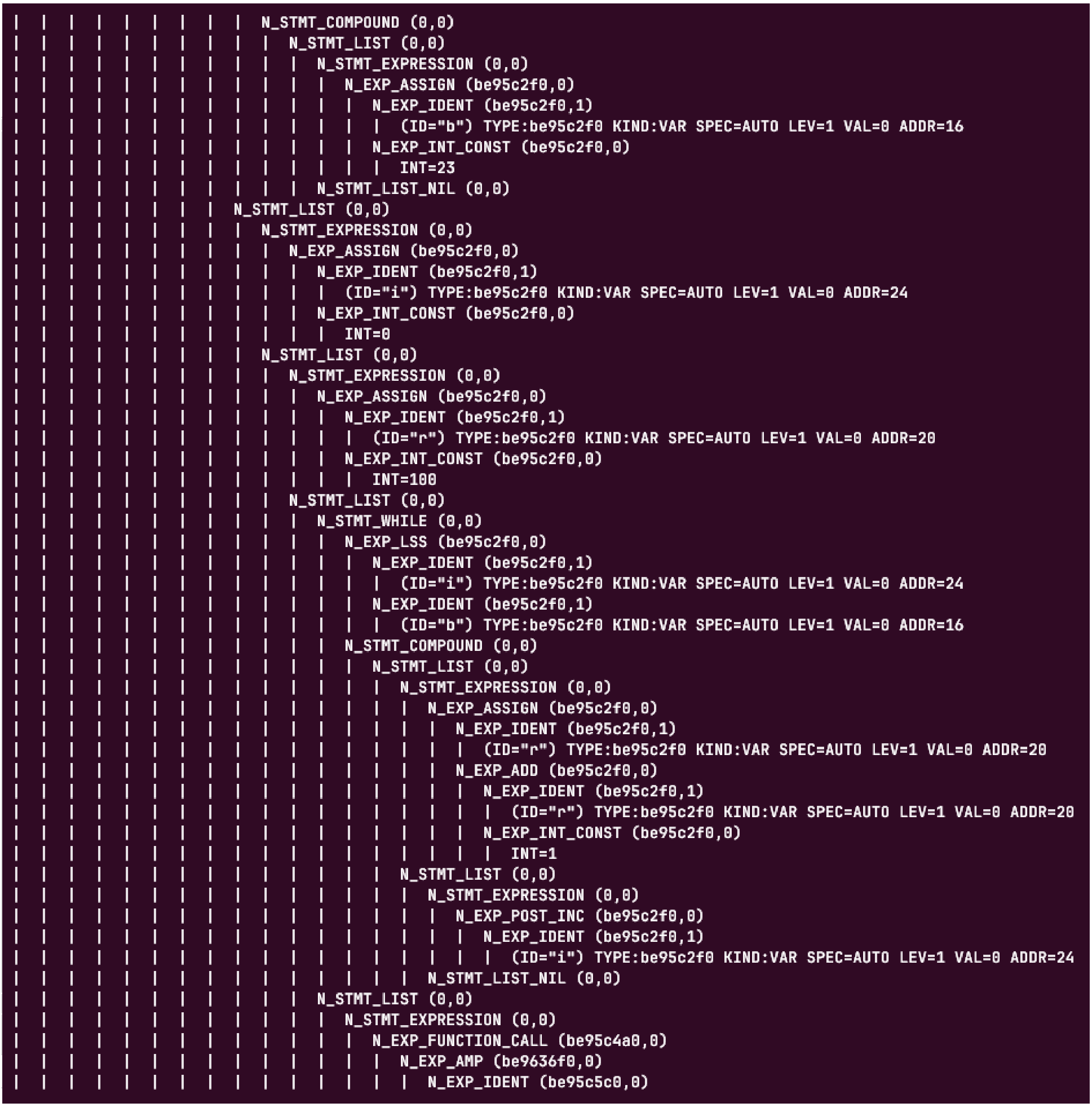


```
kh@ThinkPad-T16g2:~/compiler/hw6/cmake-build-debug$ ./a.out < ../testdir/4.c
=======  syntax tree  ==========
N_PROGRAM (0,0)
|  (ID="main") TYPE:be961980 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
|  |  |  |  |  (int)
```

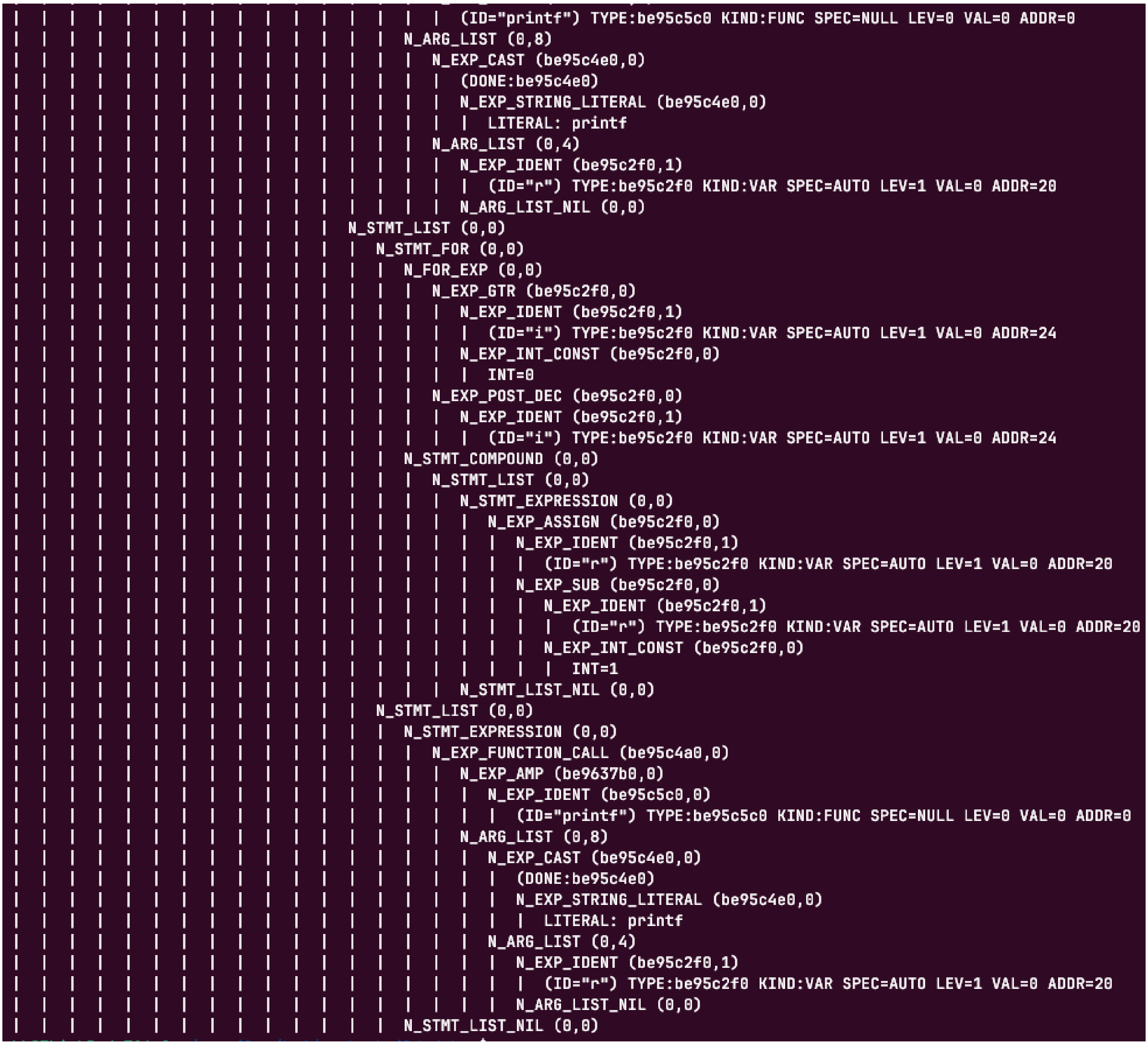
(syntax tree 생략)



```
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
=======  semantic tree  ==========
N_PROGRAM (0,28)
|  (ID="main") TYPE:be961980 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
|  |  |  |  |  (int)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,16)
|  |  |  |  |  |  (ID="a") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="b") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="r") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="i") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  N_EXP_ASSIGN (be95c2f0,0)
|  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  (ID="a") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  INT=10
|  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  N_STMT_IF_ELSE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_EQL (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_MOD (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  (ID="a") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  INT=2
|  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=11
|  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
```

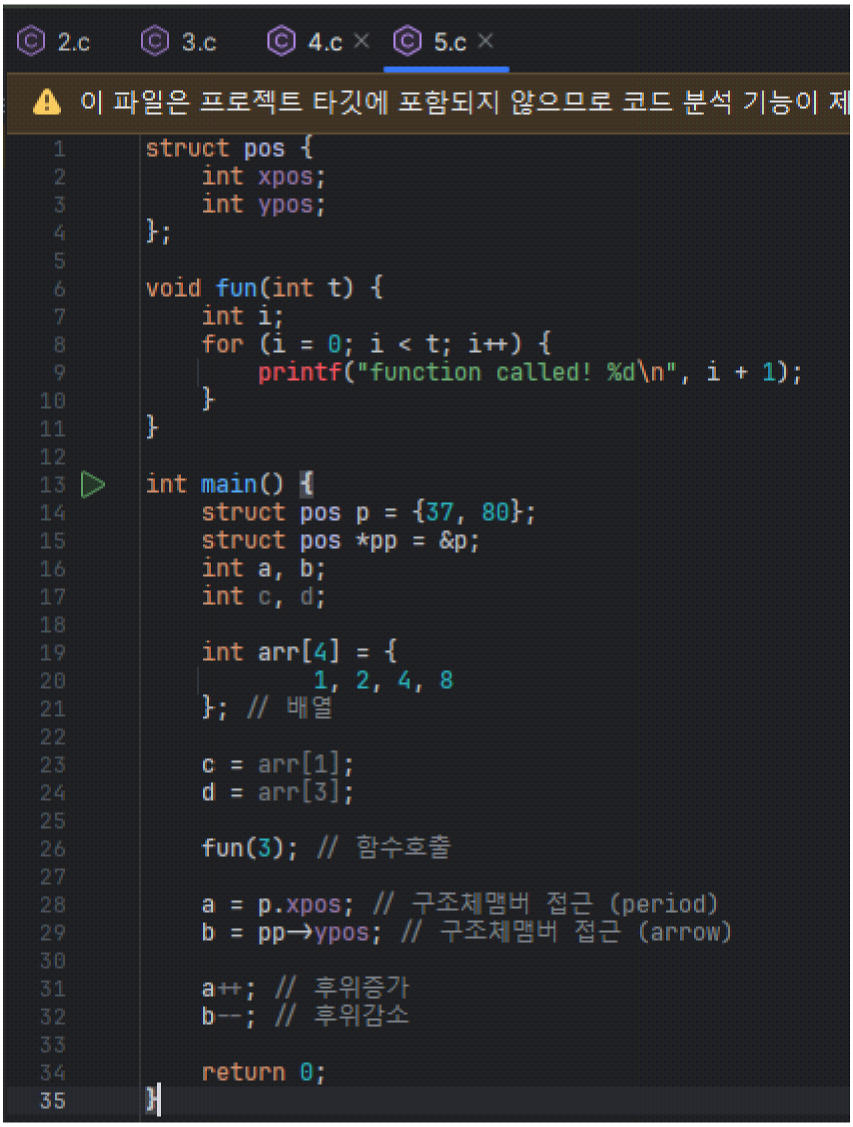


```
|  |  |  |  |  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=23
|  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  (ID="r") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  INT=100
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_STMT_WHILE (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_LSS (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="r") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ADD (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="r") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_POST_INC (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (be95c4a0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (be9636f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c5c0,0)
```



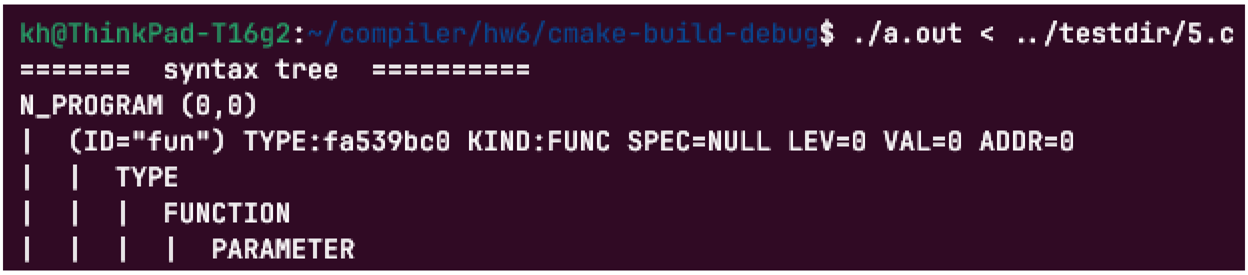
```
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="printf") TYPE:be95c5c0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,8)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (be95c4e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (DONE:be95c4e0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRING_LITERAL (be95c4e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: printf
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="r") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_FOR (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_FOR_EXP (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_GTR (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_POST_DEC (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="r") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_SUB (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="r") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (be95c2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (be95c4a0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (be9637b0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c5c0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="printf") TYPE:be95c5c0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,8)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (be95c4e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (DONE:be95c4e0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRING_LITERAL (be95c4e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: printf
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (be95c2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="r") TYPE:be95c2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
```

3-5. 5.c - postfix expression : array, function call, struct member, postfix increment/decrement



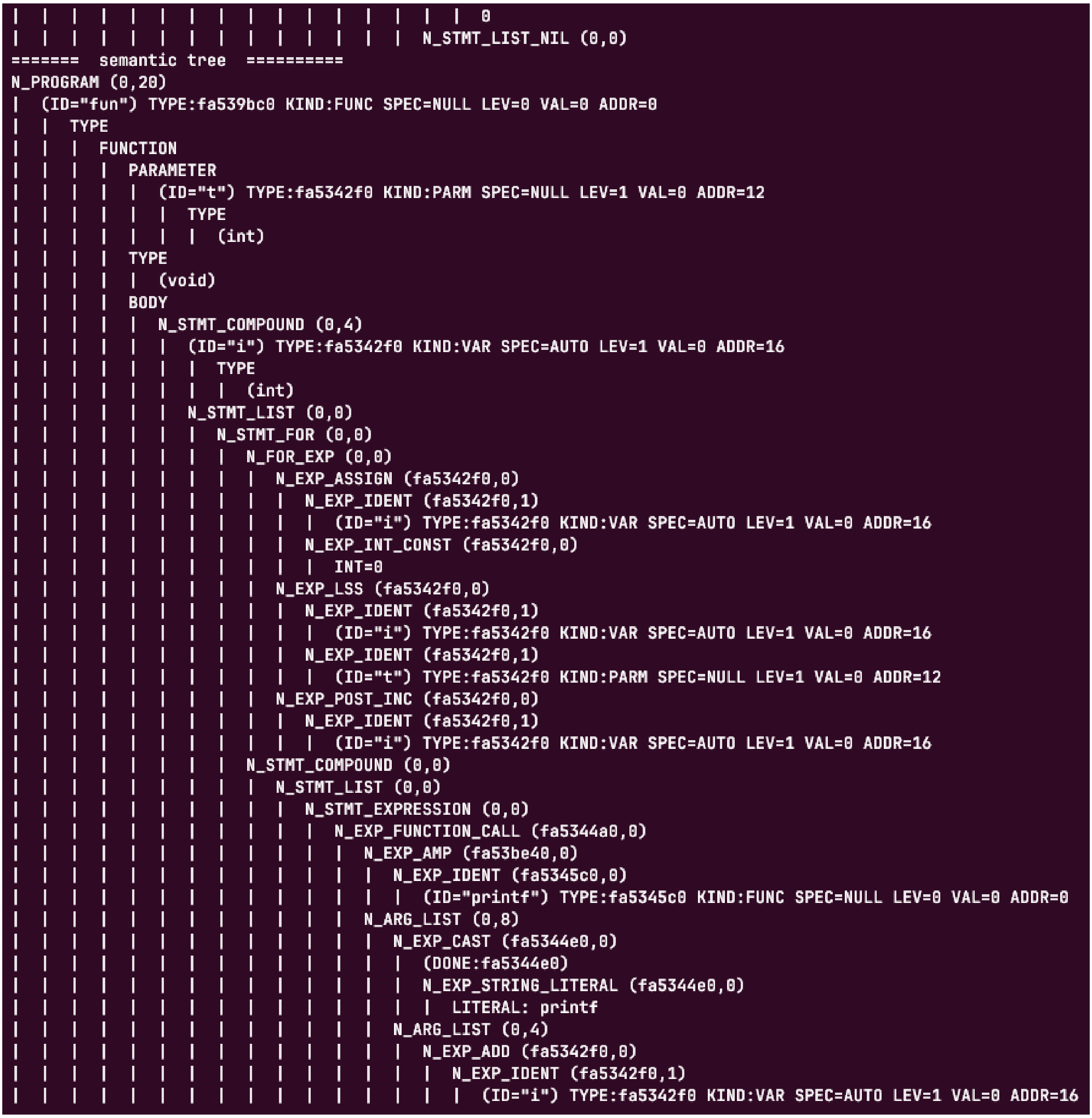
이 파일은 프로젝트 타깃에 포함되지 않으므로 코드 분석 기능이 제

```c
struct pos {
    int xpos;
    int ypos;
};

void fun(int t) {
    int i;
    for (i = 0; i < t; i++) {
        printf("function called! %d\n", i + 1);
    }
}

int main() {
    struct pos p = {37, 80};
    struct pos *pp = &p;
    int a, b;
    int c, d;

    int arr[4] = {
            1, 2, 4, 8
    }; // 배열

    c = arr[1];
    d = arr[3];

    fun(3); // 함수호출

    a = p.xpos; // 구조체맴버 접근 (period)
    b = pp->ypos; // 구조체맴버 접근 (arrow)

    a++; // 후위증가
    b--; // 후위감소

    return 0;
}
```

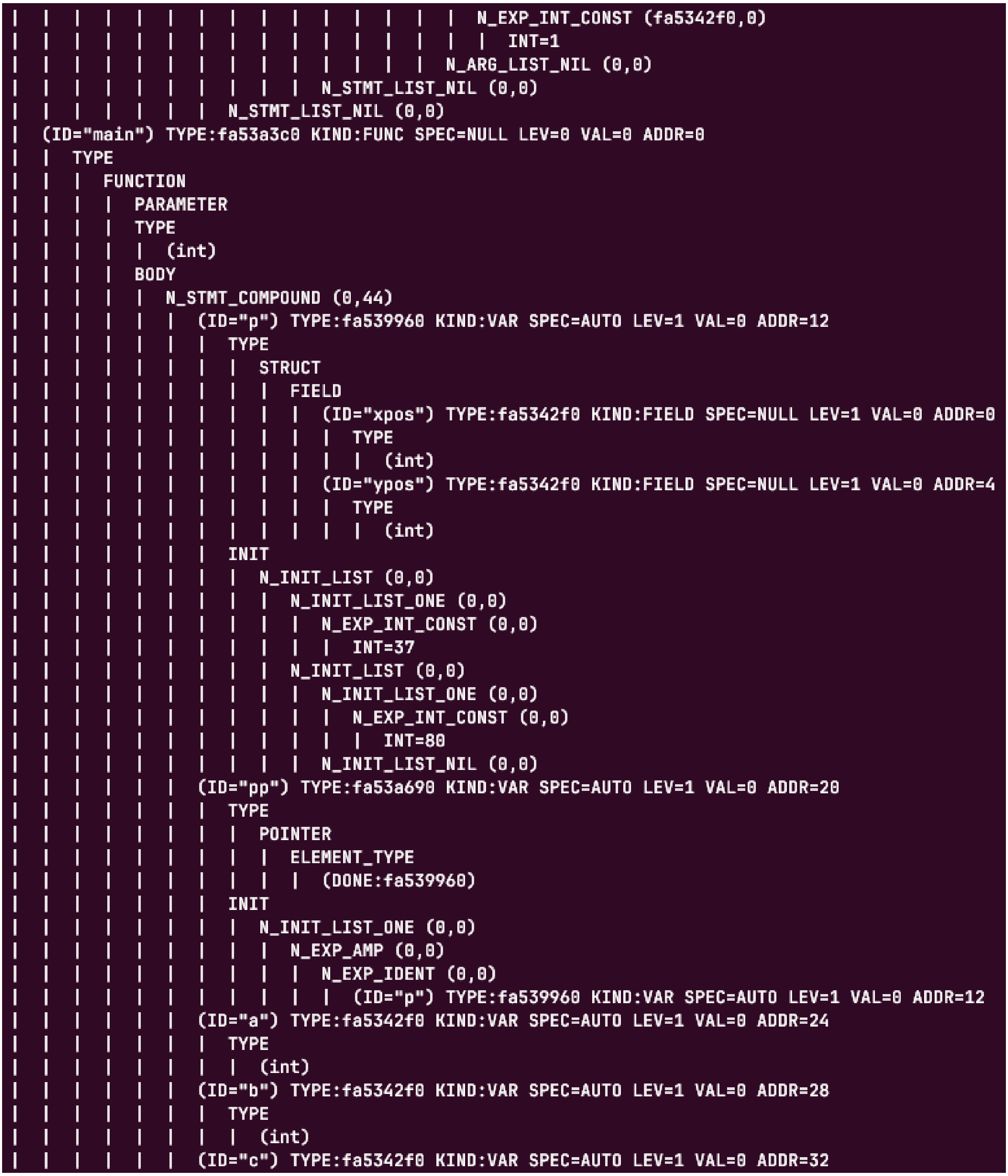


```
kh@ThinkPad-T16g2:~/compiler/hw6/cmake-build-debug$ ./a.out < ../testdir/5.c
=======  syntax tree  ==========
N_PROGRAM (0,0)
|  (ID="fun") TYPE:fa539bc0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
```

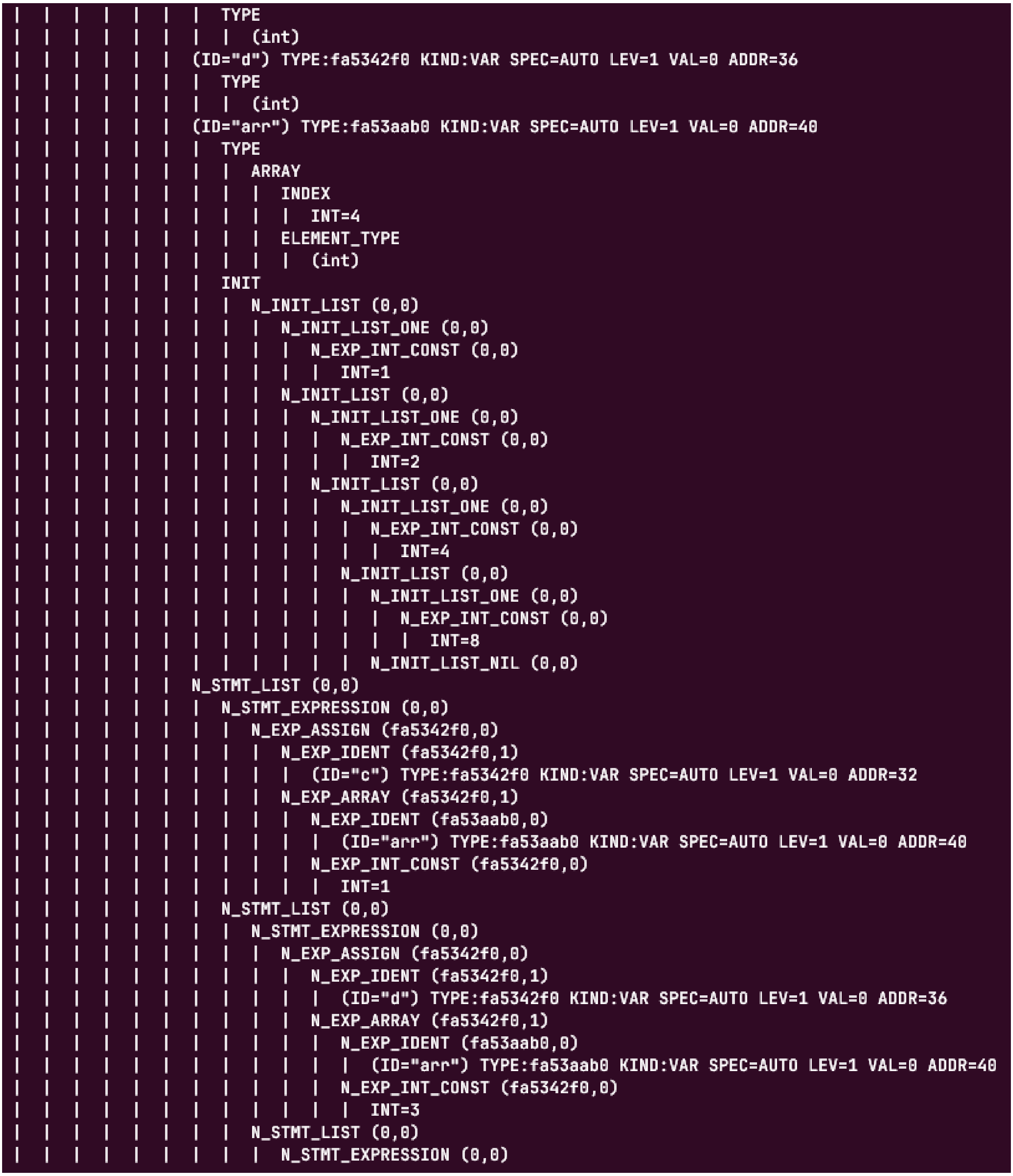
(syntax tree 생략)



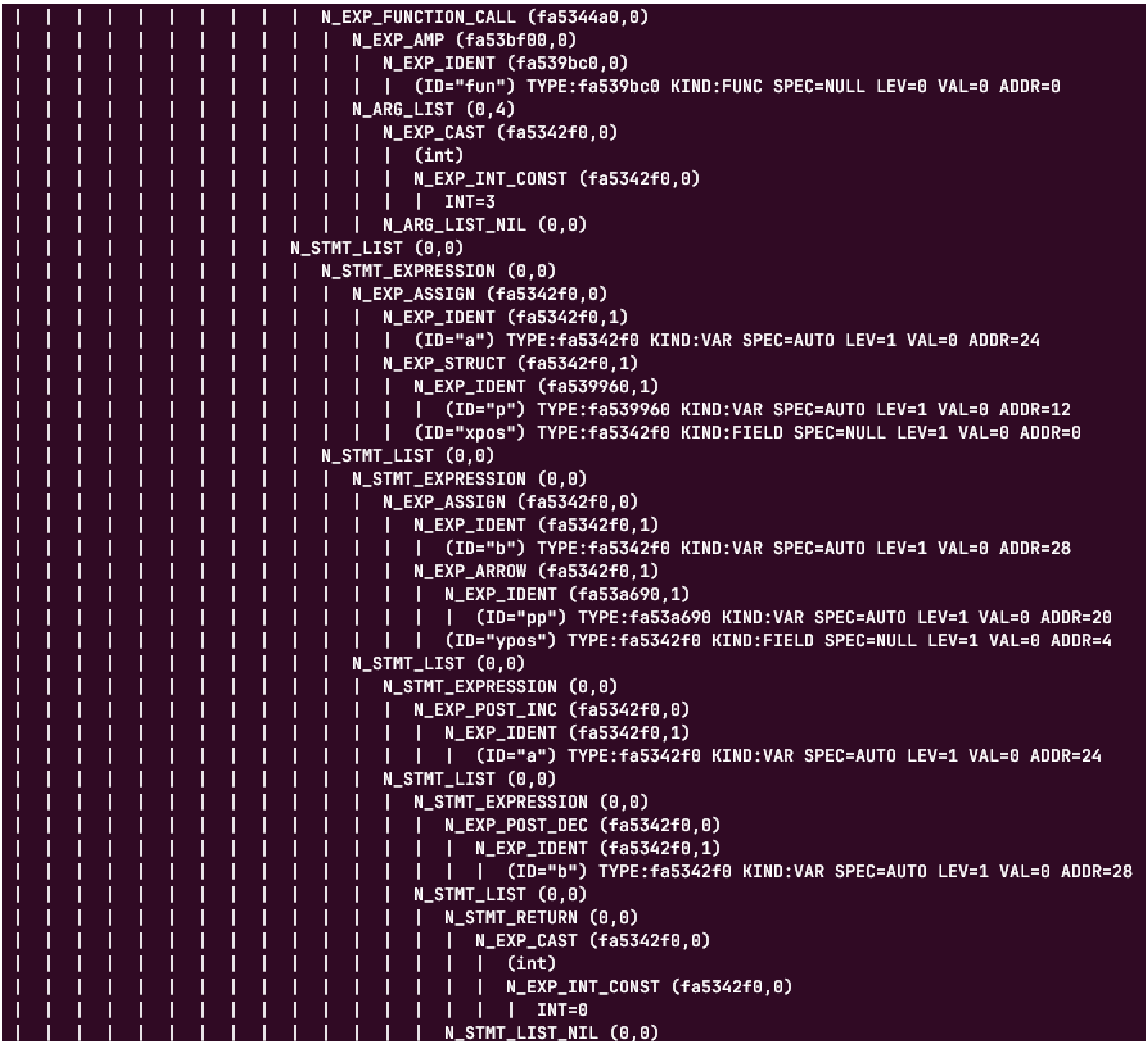
```
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
=======  semantic tree  ==========
N_PROGRAM (0,20)
|  (ID="fun") TYPE:fa539bc0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  |  (ID="t") TYPE:fa5342f0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (int)
|  |  |  |  TYPE
|  |  |  |  |  (void)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,4)
|  |  |  |  |  |  (ID="i") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_FOR (0,0)
|  |  |  |  |  |  |  |  N_FOR_EXP (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  |  |  N_EXP_LSS (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="t") TYPE:fa5342f0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  N_EXP_POST_INC (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (fa5344a0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (fa53be40,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5345c0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="printf") TYPE:fa5345c0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,8)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (fa5344e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (DONE:fa5344e0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRING_LITERAL (fa5344e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: printf
|  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ADD (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
```



```
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  (ID="main") TYPE:fa53a3c0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
|  |  |  |  |  (int)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,44)
|  |  |  |  |  |  (ID="p") TYPE:fa539960 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  STRUCT
|  |  |  |  |  |  |  |  |  FIELD
|  |  |  |  |  |  |  |  |  |  (ID="xpos") TYPE:fa5342f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  |  |  |  (ID="ypos") TYPE:fa5342f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=4
|  |  |  |  |  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  INT=37
|  |  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  INT=80
|  |  |  |  |  |  |  |  |  |  N_INIT_LIST_NIL (0,0)
|  |  |  |  |  |  (ID="pp") TYPE:fa53a690 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  POINTER
|  |  |  |  |  |  |  |  |  ELEMENT_TYPE
|  |  |  |  |  |  |  |  |  |  (DONE:fa539960)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_AMP (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="p") TYPE:fa539960 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  (ID="a") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="b") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=28
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="c") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=32
```

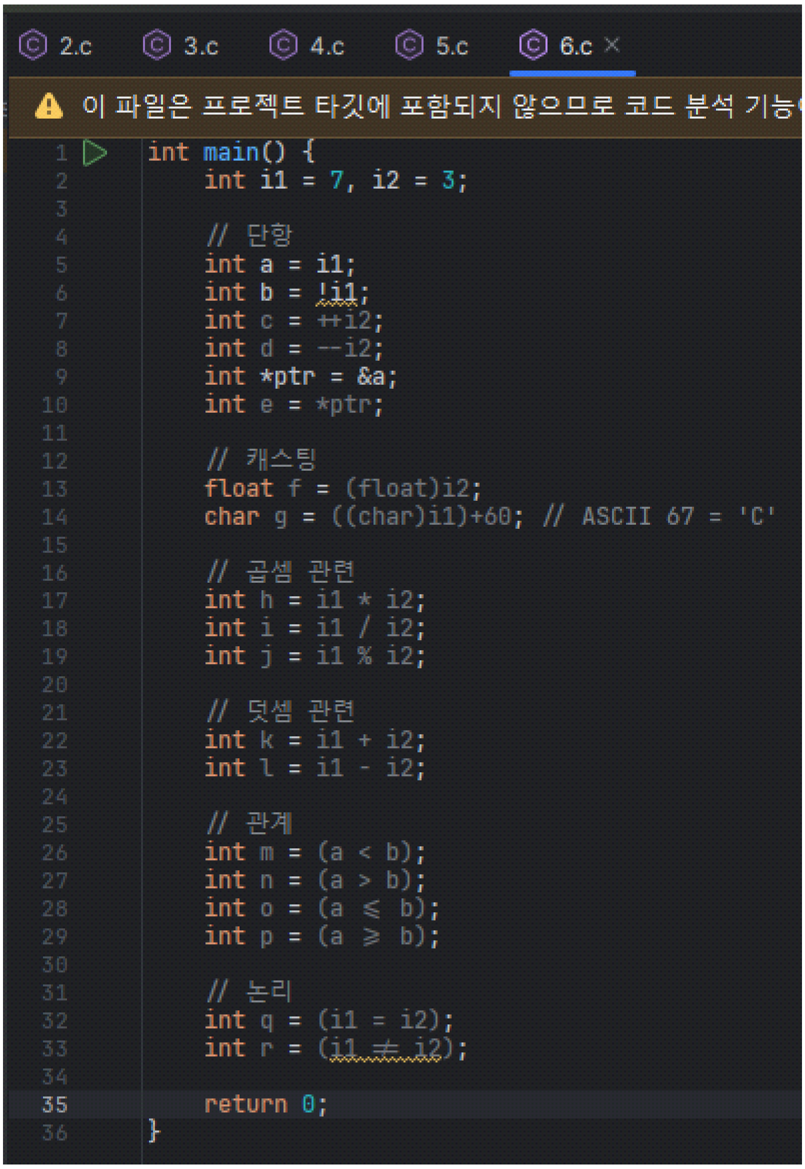


```
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="d") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=36
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="arr") TYPE:fa53aab0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=40
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  ARRAY
|  |  |  |  |  |  |  |  |  INDEX
|  |  |  |  |  |  |  |  |  |  INT=4
|  |  |  |  |  |  |  |  |  ELEMENT_TYPE
|  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  INT=2
|  |  |  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  INT=4
|  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=8
|  |  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST_NIL (0,0)
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  N_EXP_ASSIGN (fa5342f0,0)
|  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  (ID="c") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=32
|  |  |  |  |  |  |  |  |  N_EXP_ARRAY (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa53aab0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="arr") TYPE:fa53aab0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=40
|  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="d") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=36
|  |  |  |  |  |  |  |  |  |  N_EXP_ARRAY (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa53aab0,0)
|  |  |  |  |  |  |  |  |  |  |  |  (ID="arr") TYPE:fa53aab0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=40
|  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  INT=3
|  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
```



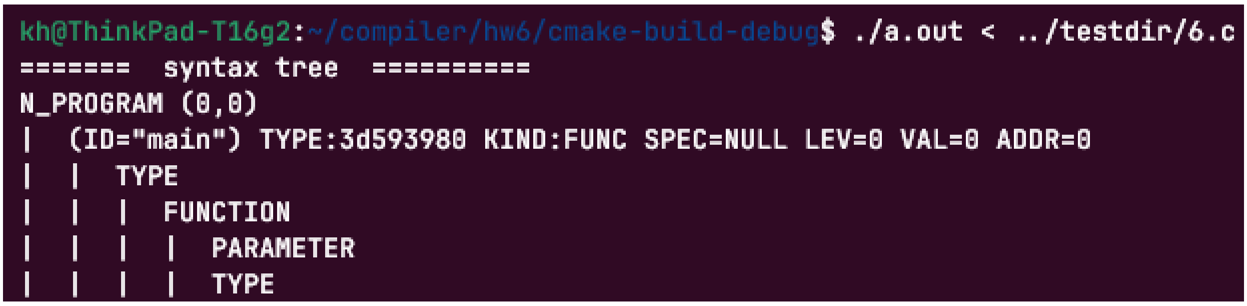
```
|  |  |  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (fa5344a0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (fa53bf00,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa539bc0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  (ID="fun") TYPE:fa539bc0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=3
|  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  (ID="a") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRUCT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa539960,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="p") TYPE:fa539960 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  |  |  |  (ID="xpos") TYPE:fa5342f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=28
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ARROW (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa53a690,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="pp") TYPE:fa53a690 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="ypos") TYPE:fa5342f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=4
|  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_POST_INC (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="a") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_POST_DEC (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (fa5342f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:fa5342f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=28
|  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_RETURN (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (fa5342f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
```

3-6. 6.c - unary expression : casting, relational, logical expression, prefix increment/decrement, pointer



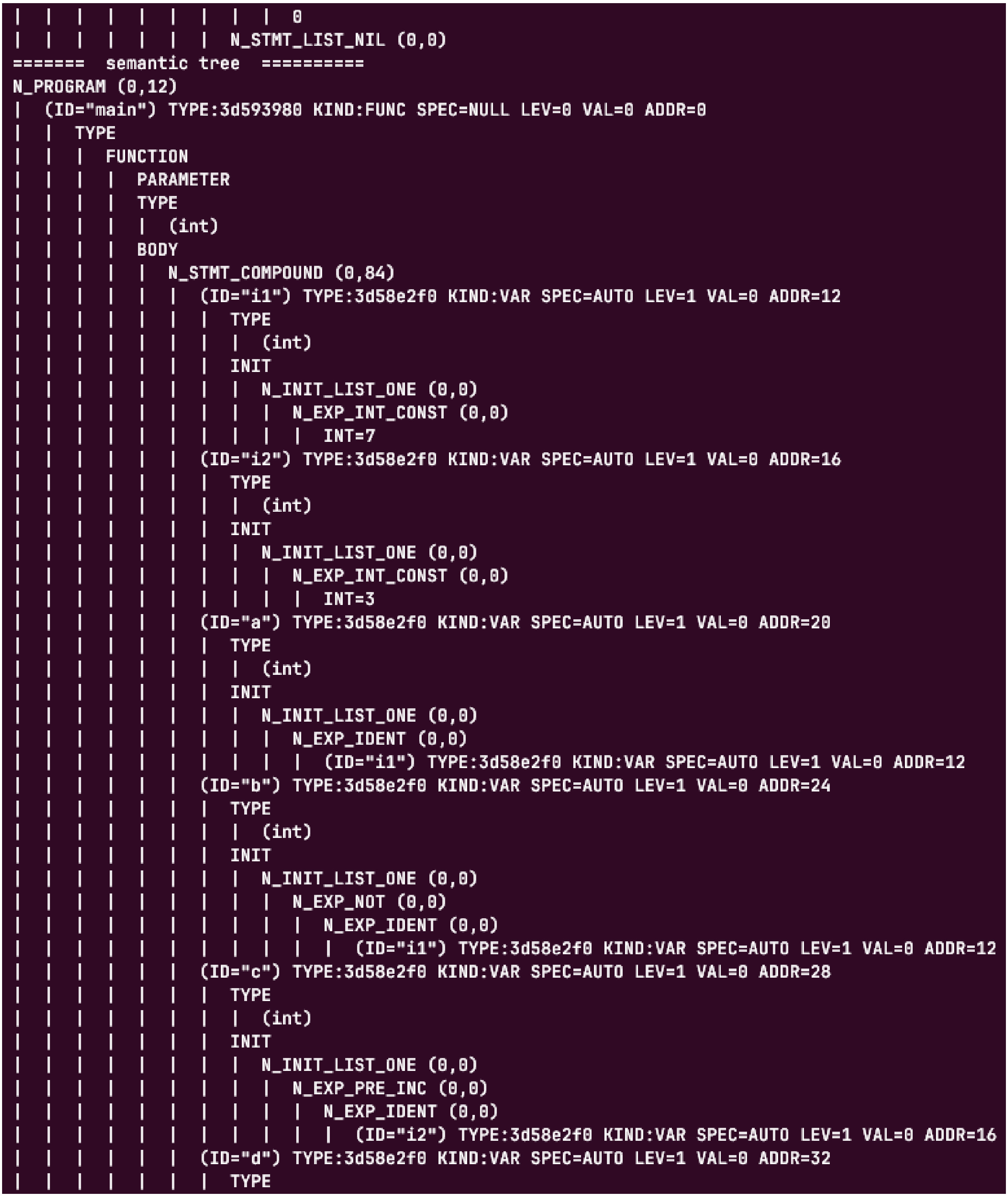
이 파일은 프로젝트 타깃에 포함되지 않으므로 코드 분석 기능

```
int main() {
    int i1 = 7, i2 = 3;

    // 단항
    int a = i1;
    int b = !i1;
    int c = ++i2;
    int d = --i2;
    int *ptr = &a;
    int e = *ptr;

    // 캐스팅
    float f = (float)i2;
    char g = ((char)i1)+60; // ASCII 67 = 'C'

    // 곱셈 관련
    int h = i1 * i2;
    int i = i1 / i2;
    int j = i1 % i2;

    // 덧셈 관련
    int k = i1 + i2;
    int l = i1 - i2;

    // 관계
    int m = (a < b);
    int n = (a > b);
    int o = (a ≤ b);
    int p = (a ≥ b);

    // 논리
    int q = (i1 = i2);
    int r = (i1 ≠ i2);

    return 0;
}
```

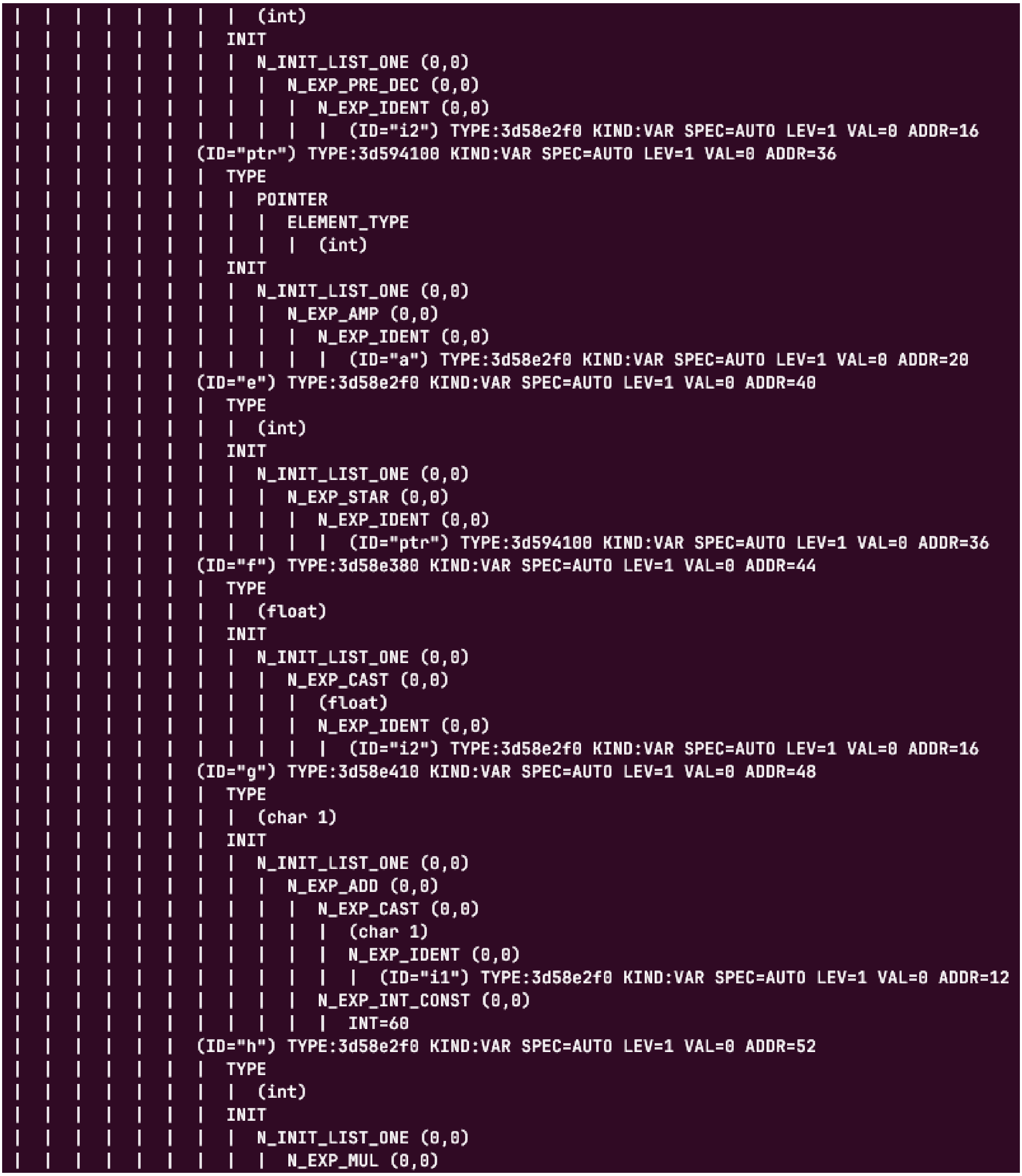


```
kh@ThinkPad-T16g2:~/compiler/hw6/cmake-build-debug$ ./a.out < ../testdir/6.c
=======  syntax tree  ==========
N_PROGRAM (0,0)
|  (ID="main") TYPE:3d593980 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
```

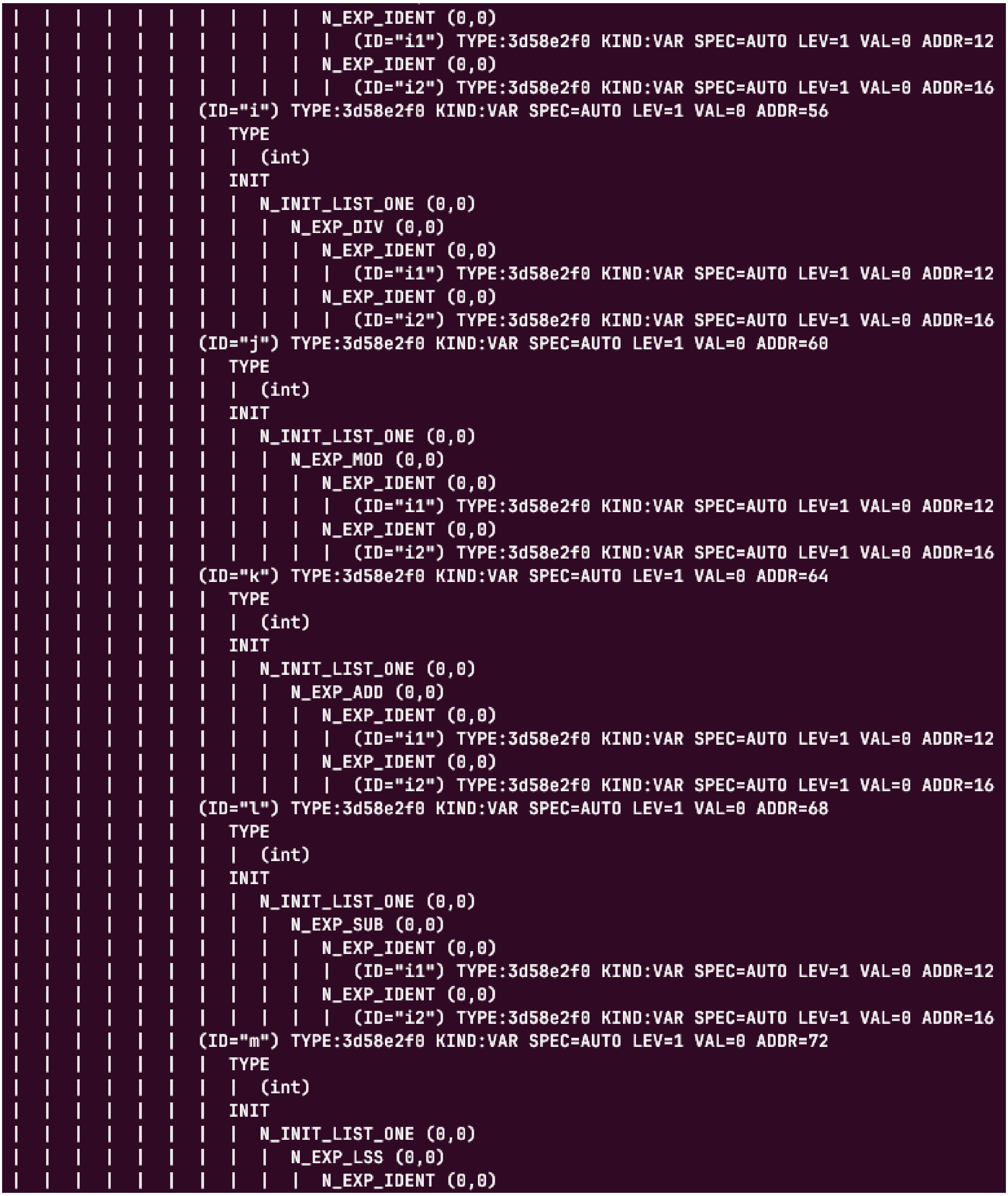
(syntax tree 생략)



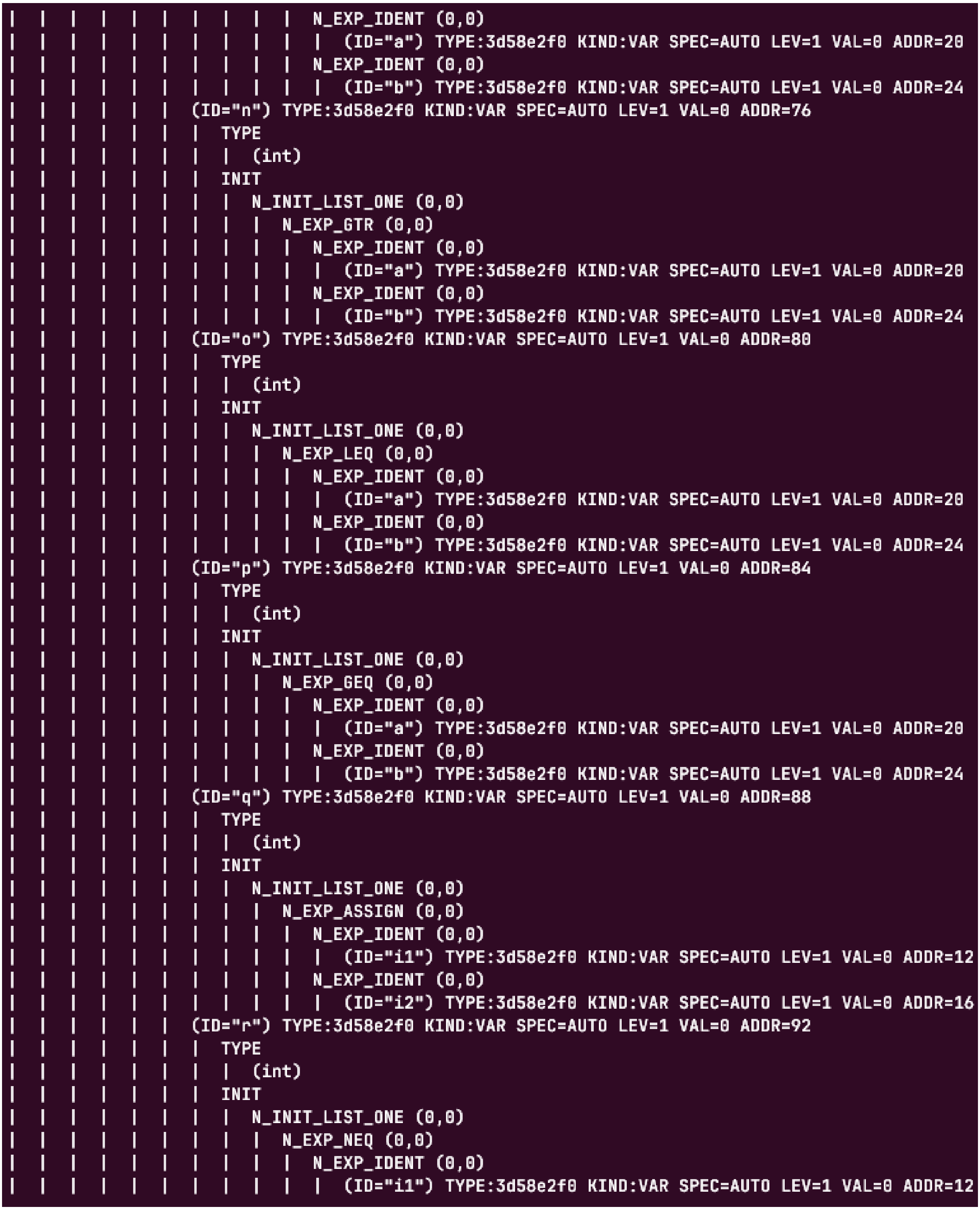
```
|  |  |  |  |  |  |  |  |  0
|  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
=======  semantic tree  ==========
N_PROGRAM (0,12)
|  (ID="main") TYPE:3d593980 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
|  |  |  |  |  (int)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,84)
|  |  |  |  |  |  (ID="i1") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  INT=7
|  |  |  |  |  |  (ID="i2") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  INT=3
|  |  |  |  |  |  (ID="a") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  (ID="i1") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  (ID="b") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_NOT (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i1") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  (ID="c") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=28
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_PRE_INC (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i2") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  (ID="d") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=32
|  |  |  |  |  |  |  TYPE
```



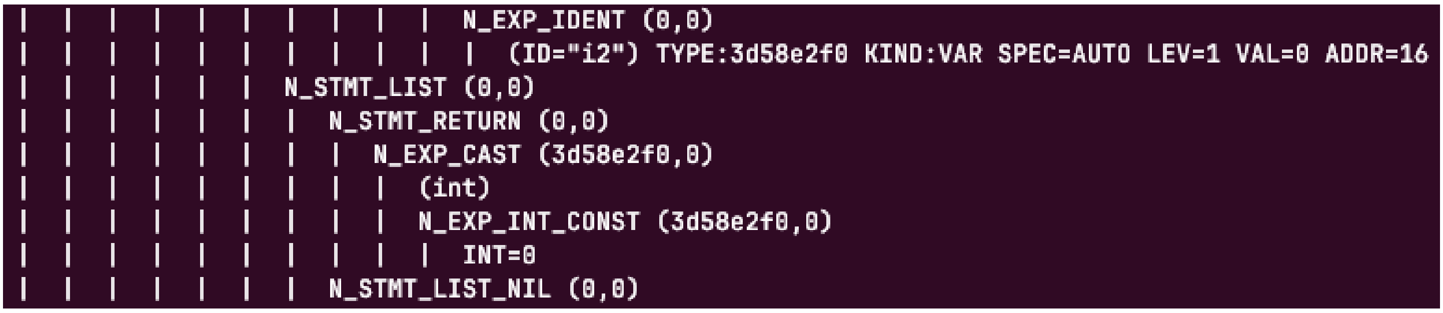
```
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_PRE_DEC (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i2") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  (ID="ptr") TYPE:3d594100 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=36
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  POINTER
|  |  |  |  |  |  |  |  |  ELEMENT_TYPE
|  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_AMP (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="a") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  (ID="e") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=40
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_STAR (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="ptr") TYPE:3d594100 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=36
|  |  |  |  |  |  (ID="f") TYPE:3d58e380 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=44
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (float)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_CAST (0,0)
|  |  |  |  |  |  |  |  |  |  (float)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i2") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  (ID="g") TYPE:3d58e410 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=48
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (char 1)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ADD (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_CAST (0,0)
|  |  |  |  |  |  |  |  |  |  |  (char 1)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  (ID="i1") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  INT=60
|  |  |  |  |  |  (ID="h") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=52
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_MUL (0,0)
```



```
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i1") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i2") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  (ID="i") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=56
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_DIV (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i1") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i2") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  (ID="j") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=60
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_MOD (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i1") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i2") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  (ID="k") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=64
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ADD (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i1") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i2") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  (ID="l") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=68
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_SUB (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i1") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i2") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  (ID="m") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=72
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_LSS (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
```

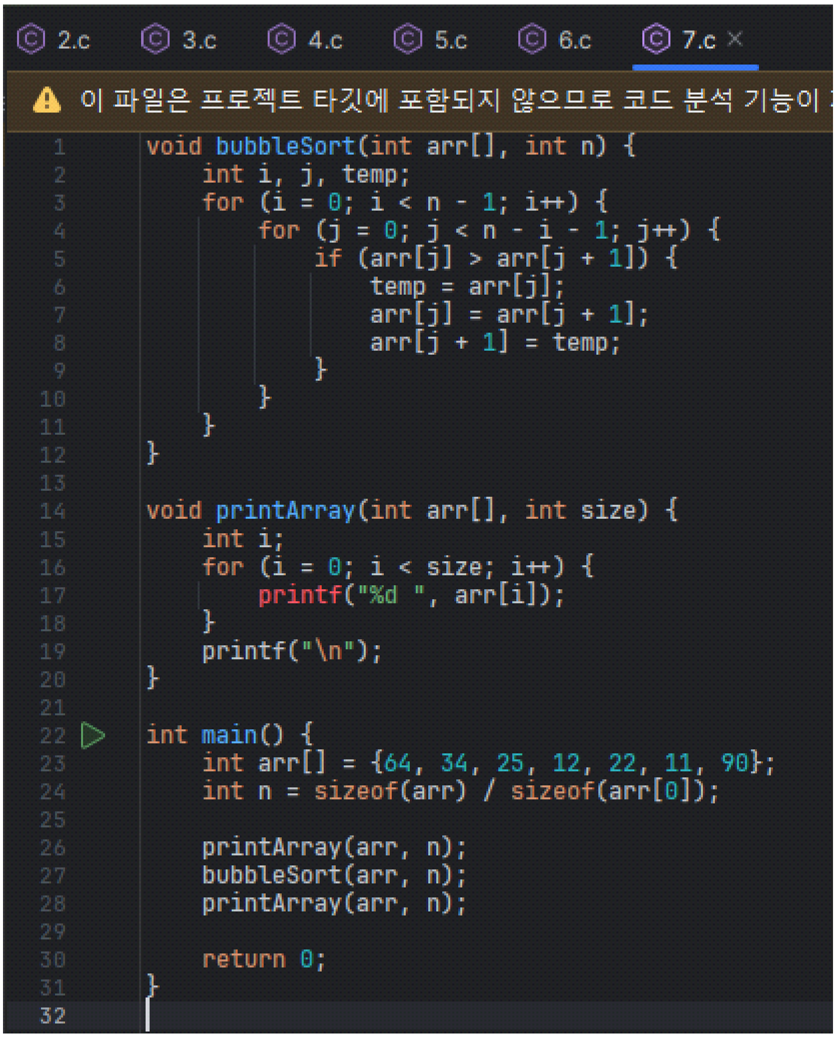


```
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="a") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  (ID="n") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=76
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_GTR (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="a") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  (ID="o") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=80
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_LEQ (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="a") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  (ID="p") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=84
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_GEQ (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="a") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="b") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  (ID="q") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=88
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i1") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i2") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  (ID="r") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=92
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_NEQ (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i1") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
```



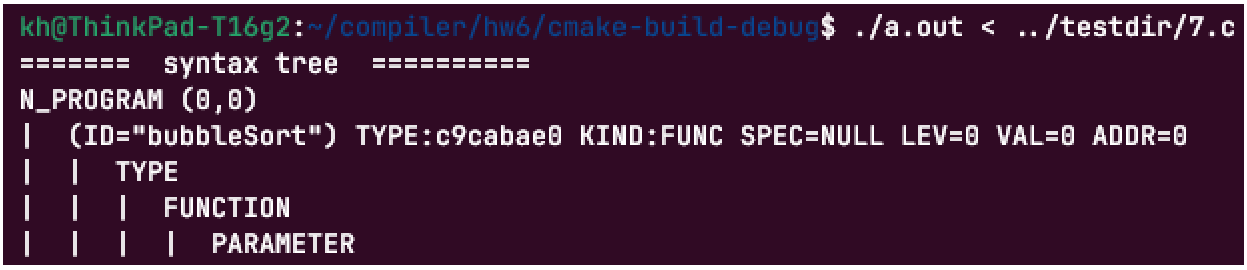
```
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (0,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="i2") TYPE:3d58e2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_RETURN (0,0)
|  |  |  |  |  |  |  |  N_EXP_CAST (3d58e2f0,0)
|  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (3d58e2f0,0)
|  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
```

3-7. 7.c - 버블소트



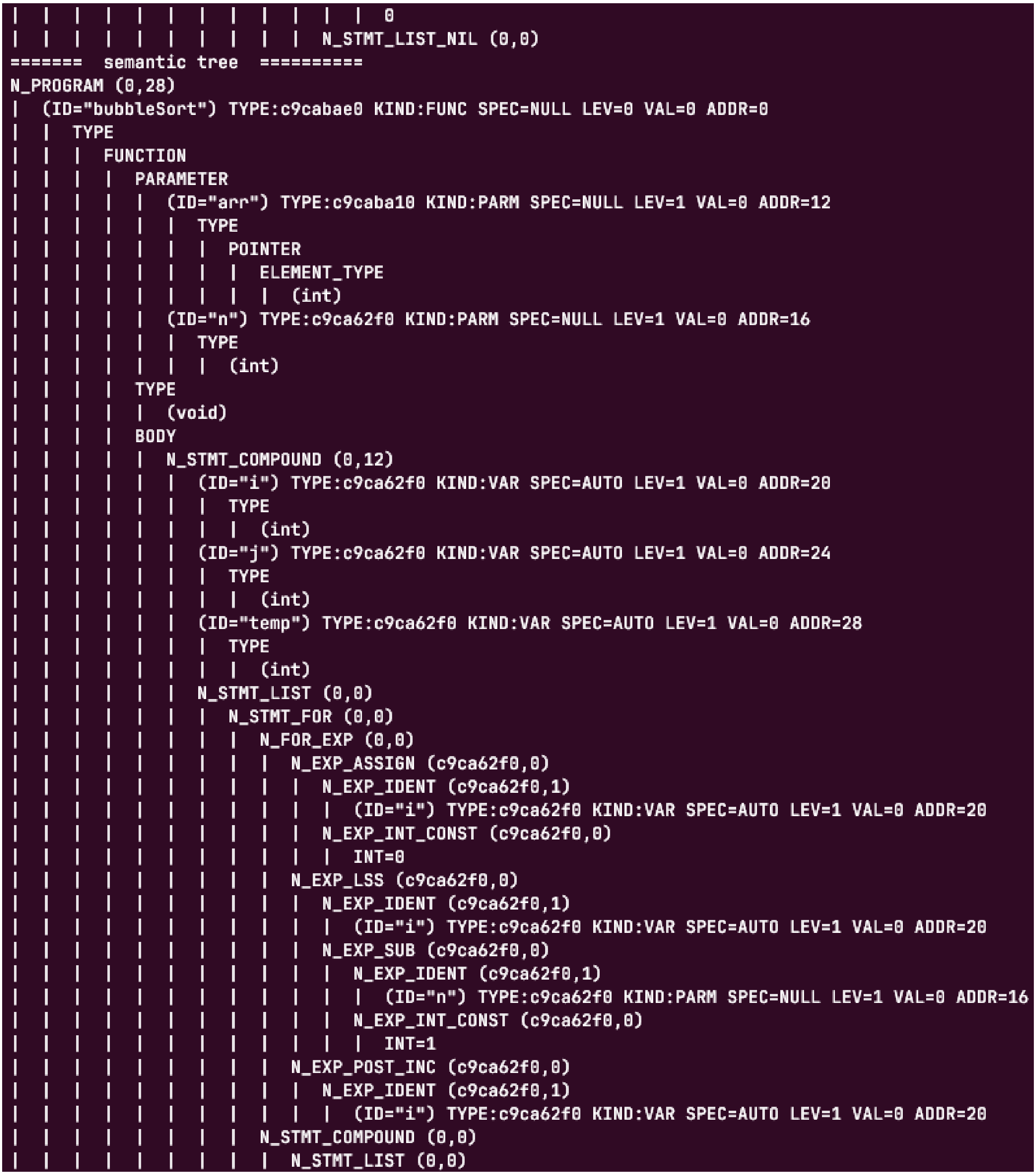
⚠ 이 파일은 프로젝트 타깃에 포함되지 않으므로 코드 분석 기능이

```c
void bubbleSort(int arr[], int n) {
    int i, j, temp;
    for (i = 0; i < n - 1; i++) {
        for (j = 0; j < n - i - 1; j++) {
            if (arr[j] > arr[j + 1]) {
                temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
}

void printArray(int arr[], int size) {
    int i;
    for (i = 0; i < size; i++) {
        printf("%d ", arr[i]);
    }
    printf("\n");
}

int main() {
    int arr[] = {64, 34, 25, 12, 22, 11, 90};
    int n = sizeof(arr) / sizeof(arr[0]);

    printArray(arr, n);
    bubbleSort(arr, n);
    printArray(arr, n);

    return 0;
}

```



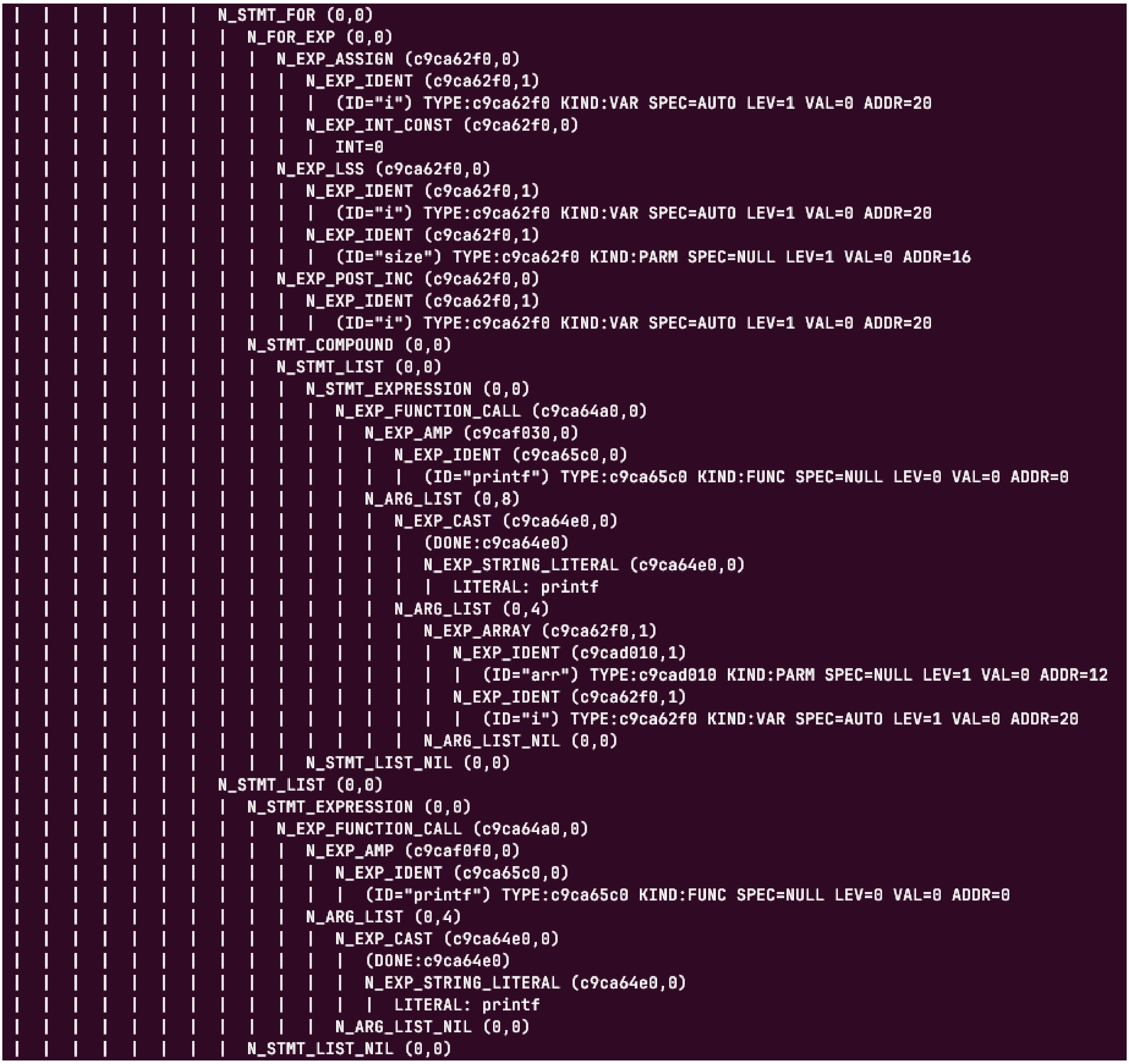
```
kh@ThinkPad-T16g2:~/compiler/hw6/cmake-build-debug$ ./a.out < ../testdir/7.c
=======  syntax tree  ==========
N_PROGRAM (0,0)
|  (ID="bubbleSort") TYPE:c9cabae0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
```

(syntax tree 생략)

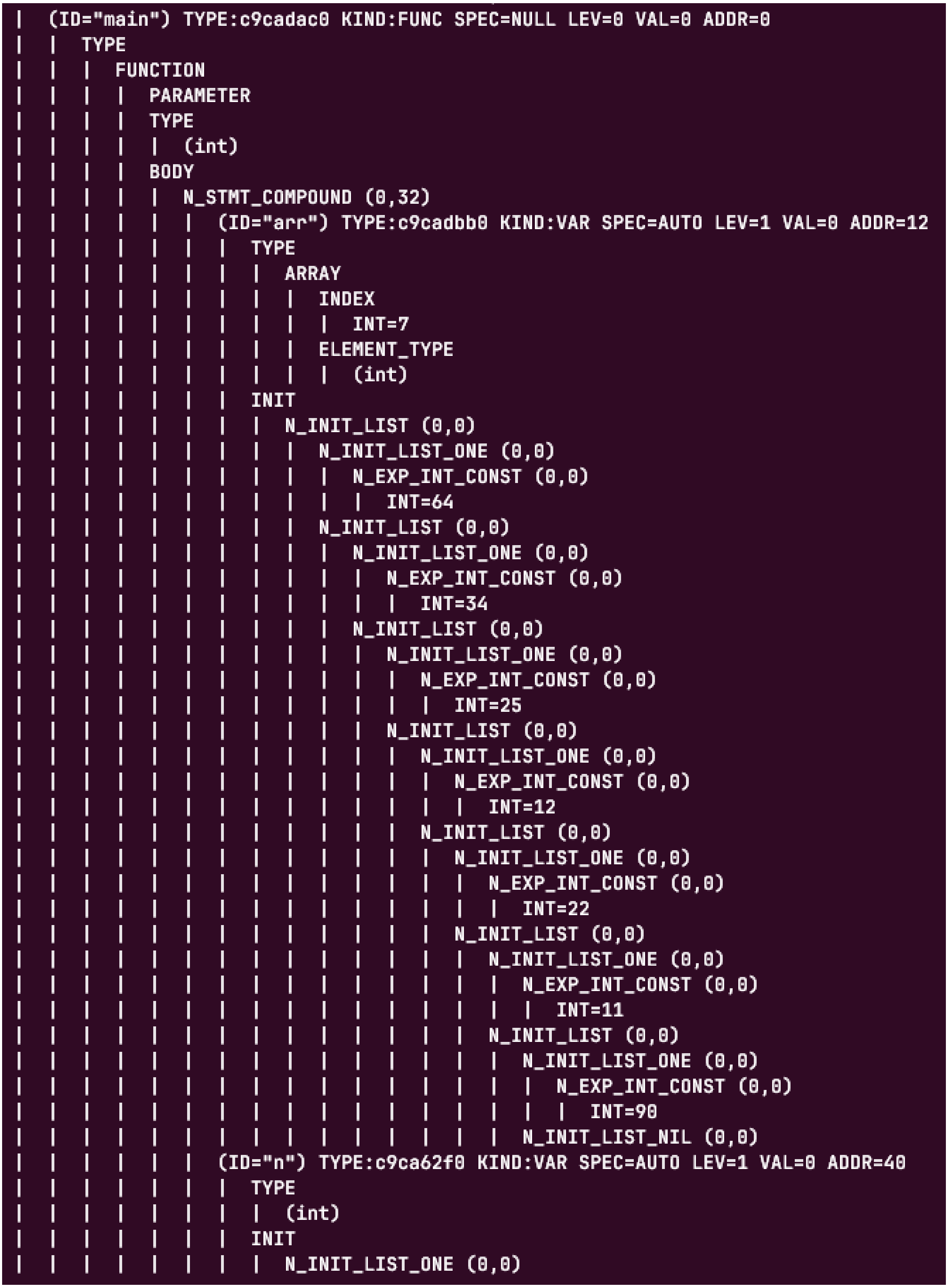


```
|  |  |  |  |  |  |  |  |  |  |  |  0
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
=======  semantic tree  ==========
N_PROGRAM (0,28)
|  (ID="bubbleSort") TYPE:c9cabae0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  |  (ID="arr") TYPE:c9caba10 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  POINTER
|  |  |  |  |  |  |  |  ELEMENT_TYPE
|  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  (ID="n") TYPE:c9ca62f0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (int)
|  |  |  |  TYPE
|  |  |  |  |  (void)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,12)
|  |  |  |  |  |  (ID="i") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="j") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=24
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="temp") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=28
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_FOR (0,0)
|  |  |  |  |  |  |  |  N_FOR_EXP (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca62f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  |  |  N_EXP_LSS (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca62f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  N_EXP_SUB (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca62f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  (ID="n") TYPE:c9ca62f0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  |  |  N_EXP_POST_INC (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca62f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
```

(semantic tree 일부 생략)

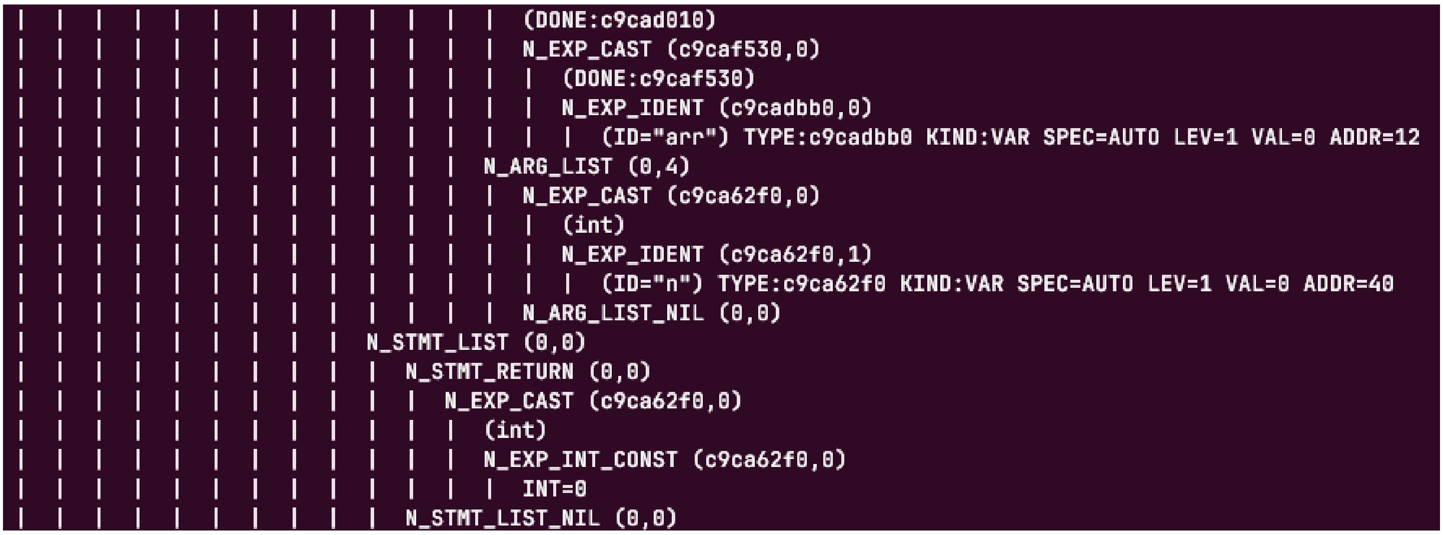


```
|  |  |  |  |  |  |  N_STMT_FOR (0,0)
|  |  |  |  |  |  |  |  N_FOR_EXP (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca62f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  |  |  N_EXP_LSS (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca62f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca62f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="size") TYPE:c9ca62f0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  N_EXP_POST_INC (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca62f0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (c9ca64a0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (c9caf030,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca65c0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="printf") TYPE:c9ca65c0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,8)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (c9ca64e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (DONE:c9ca64e0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRING_LITERAL (c9ca64e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: printf
|  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_ARRAY (c9ca62f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9cad010,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="arr") TYPE:c9cad010 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca62f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=20
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (c9ca64a0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_AMP (c9caf0f0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca65c0,0)
|  |  |  |  |  |  |  |  |  |  |  |  (ID="printf") TYPE:c9ca65c0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (c9ca64e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  (DONE:c9ca64e0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRING_LITERAL (c9ca64e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: printf
|  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
```



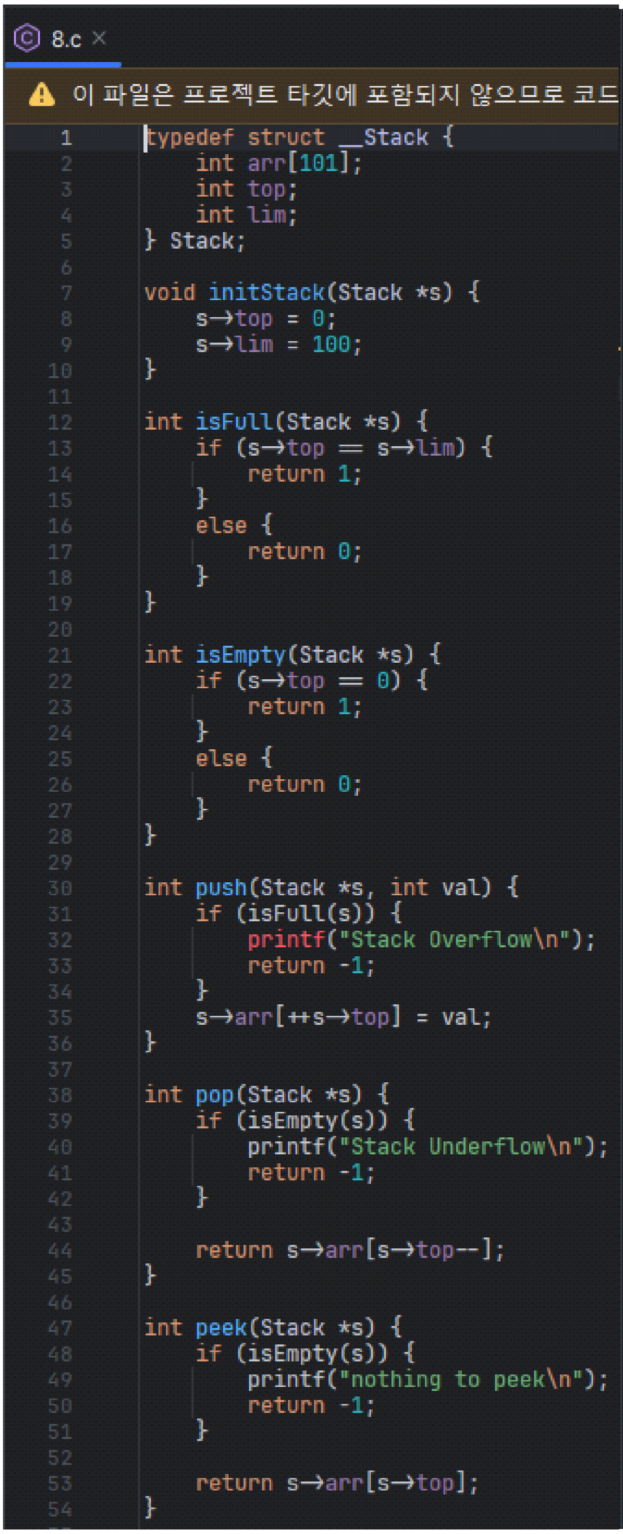
```
|  (ID="main") TYPE:c9cadac0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
|  |  |  |  |  (int)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,32)
|  |  |  |  |  |  (ID="arr") TYPE:c9cadbb0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  ARRAY
|  |  |  |  |  |  |  |  |  INDEX
|  |  |  |  |  |  |  |  |  |  INT=7
|  |  |  |  |  |  |  |  |  ELEMENT_TYPE
|  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  INT=64
|  |  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  INT=34
|  |  |  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  INT=25
|  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=12
|  |  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=22
|  |  |  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=11
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=90
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_INIT_LIST_NIL (0,0)
|  |  |  |  |  |  (ID="n") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=40
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  INIT
|  |  |  |  |  |  |  |  N_INIT_LIST_ONE (0,0)
```

(semantic tree 일부 생략)

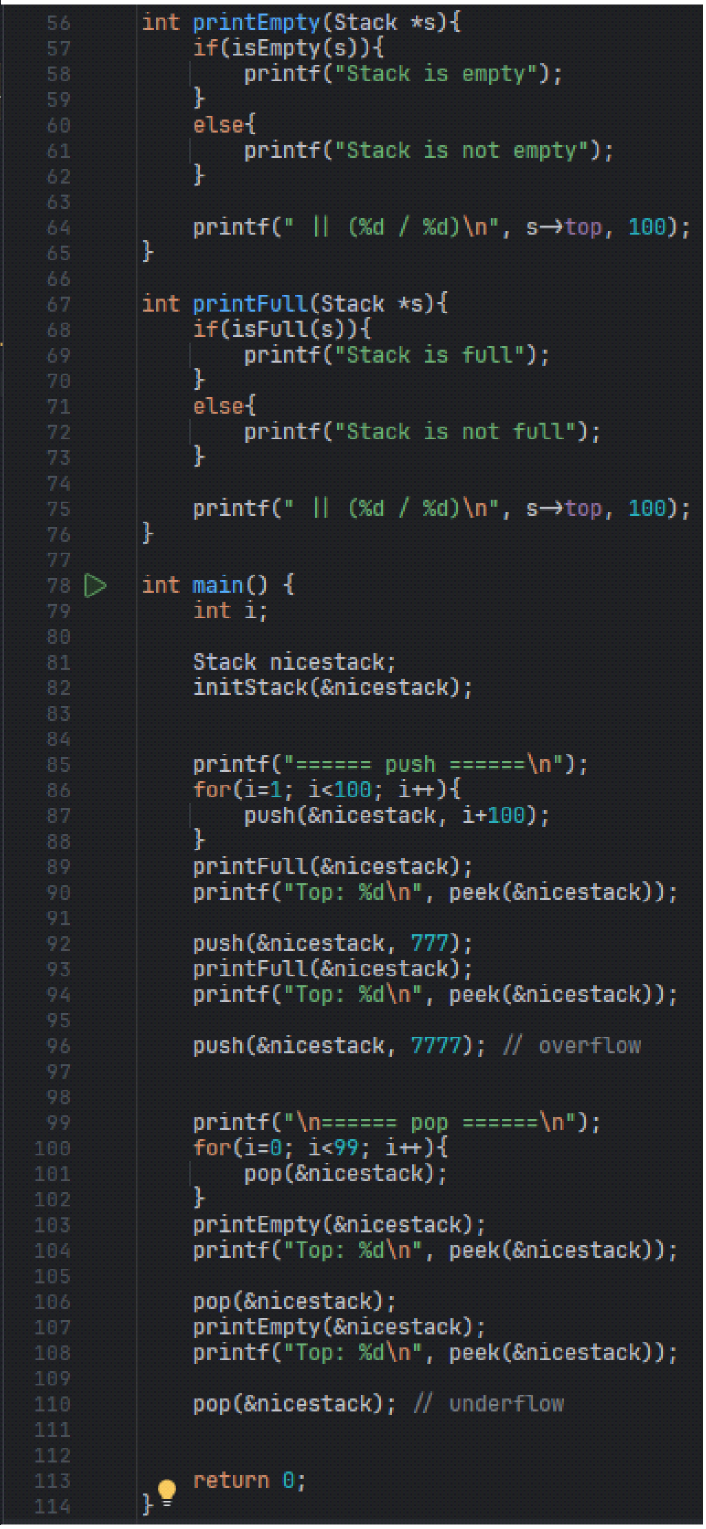


```
|  |  |  |  |  |  |  |  |  |  |  |  |  (DONE:c9cad010)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (c9caf530,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (DONE:c9caf530)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9cadbb0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="arr") TYPE:c9cadbb0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (c9ca62f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="n") TYPE:c9ca62f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=40
|  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_RETURN (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (c9ca62f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
```

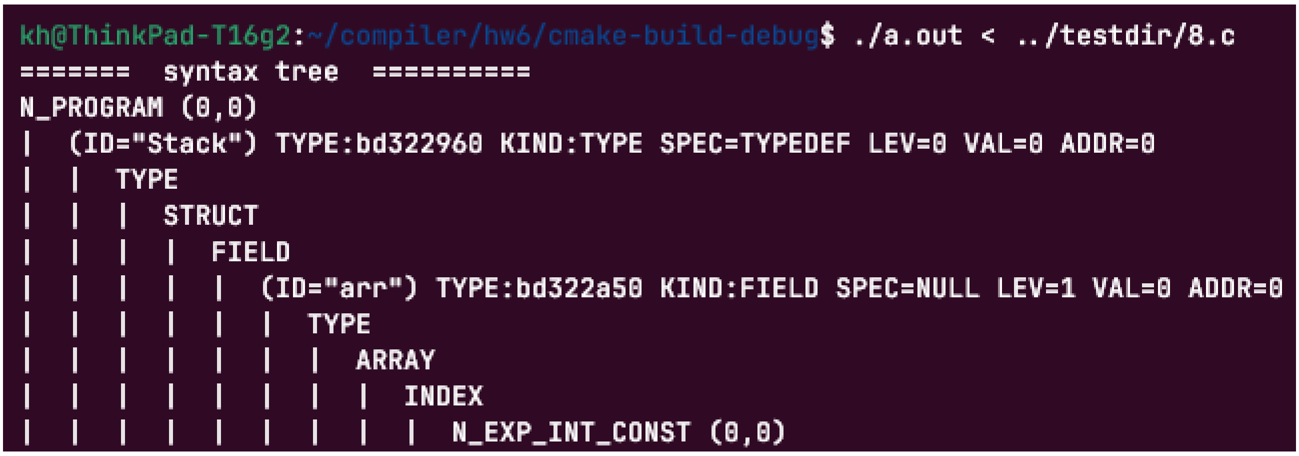
3-8. 8.c - 스택 자료구조 구현



```c
typedef struct __Stack {
    int arr[101];
    int top;
    int lim;
} Stack;

void initStack(Stack *s) {
    s->top = 0;
    s->lim = 100;
}

int isFull(Stack *s) {
    if (s->top == s->lim) {
        return 1;
    }
    else {
        return 0;
    }
}

int isEmpty(Stack *s) {
    if (s->top == 0) {
        return 1;
    }
    else {
        return 0;
    }
}

int push(Stack *s, int val) {
    if (isFull(s)) {
        printf("Stack Overflow\n");
        return -1;
    }
    s->arr[++s->top] = val;
}

int pop(Stack *s) {
    if (isEmpty(s)) {
        printf("Stack Underflow\n");
        return -1;
    }

    return s->arr[s->top--];
}

int peek(Stack *s) {
    if (isEmpty(s)) {
        printf("nothing to peek\n");
        return -1;
    }

    return s->arr[s->top];
}
```

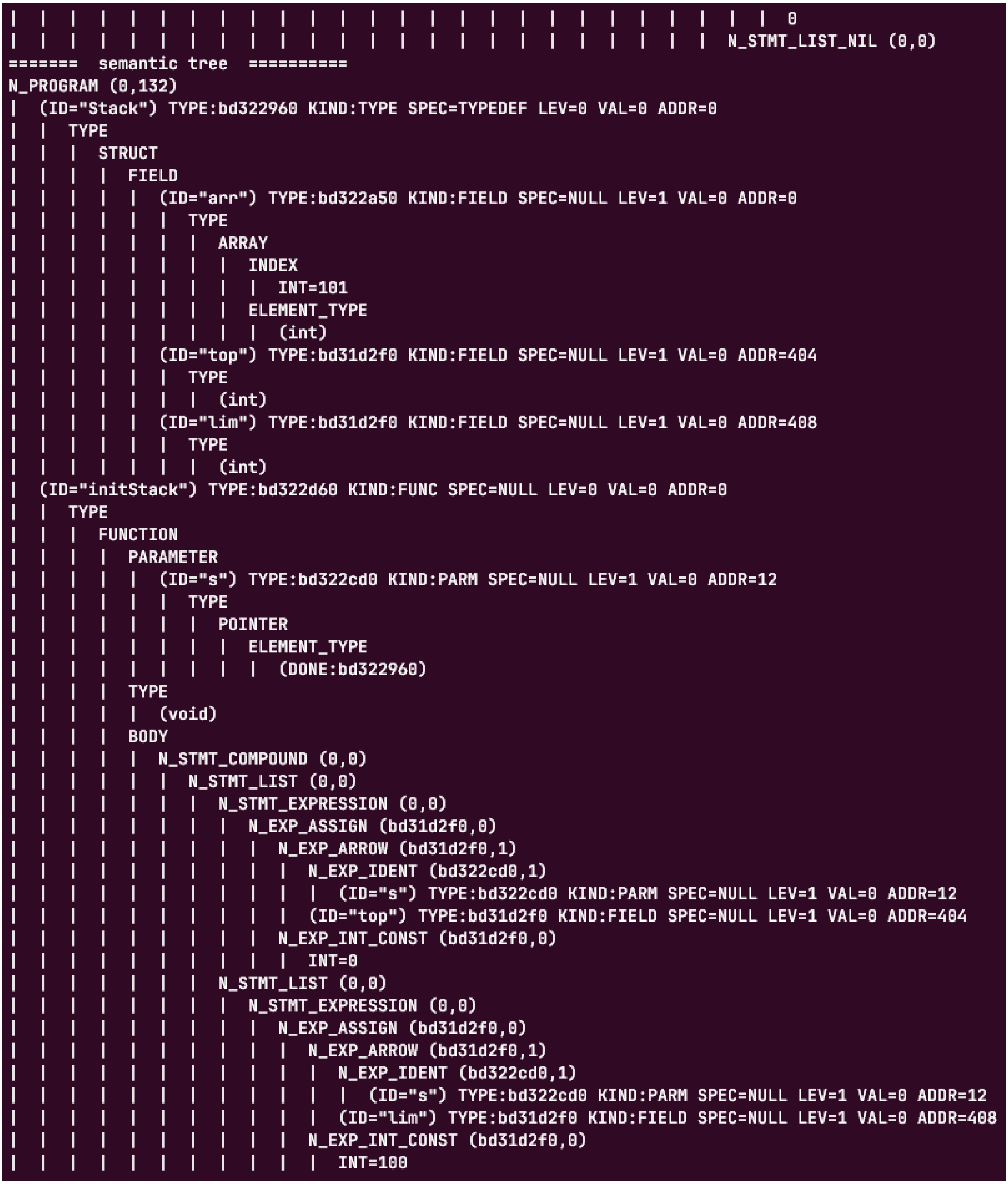


```c
int printEmpty(Stack *s){
    if(isEmpty(s)){
        printf("Stack is empty");
    }
    else{
        printf("Stack is not empty");
    }

    printf(" || (%d / %d)\n", s->top, 100);
}

int printFull(Stack *s){
    if(isFull(s)){
        printf("Stack is full");
    }
    else{
        printf("Stack is not full");
    }

    printf(" || (%d / %d)\n", s->top, 100);
}

int main() {
    int i;

    Stack nicestack;
    initStack(&nicestack);


    printf("====== push ======\n");
    for(i=1; i<100; i++){
        push(&nicestack, i+100);
    }
    printFull(&nicestack);
    printf("Top: %d\n", peek(&nicestack));

    push(&nicestack, 777);
    printFull(&nicestack);
    printf("Top: %d\n", peek(&nicestack));

    push(&nicestack, 7777); // overflow


    printf("\n====== pop ======\n");
    for(i=0; i<99; i++){
        pop(&nicestack);
    }
    printEmpty(&nicestack);
    printf("Top: %d\n", peek(&nicestack));

    pop(&nicestack);
    printEmpty(&nicestack);
    printf("Top: %d\n", peek(&nicestack));

    pop(&nicestack); // underflow


    return 0;
}
```

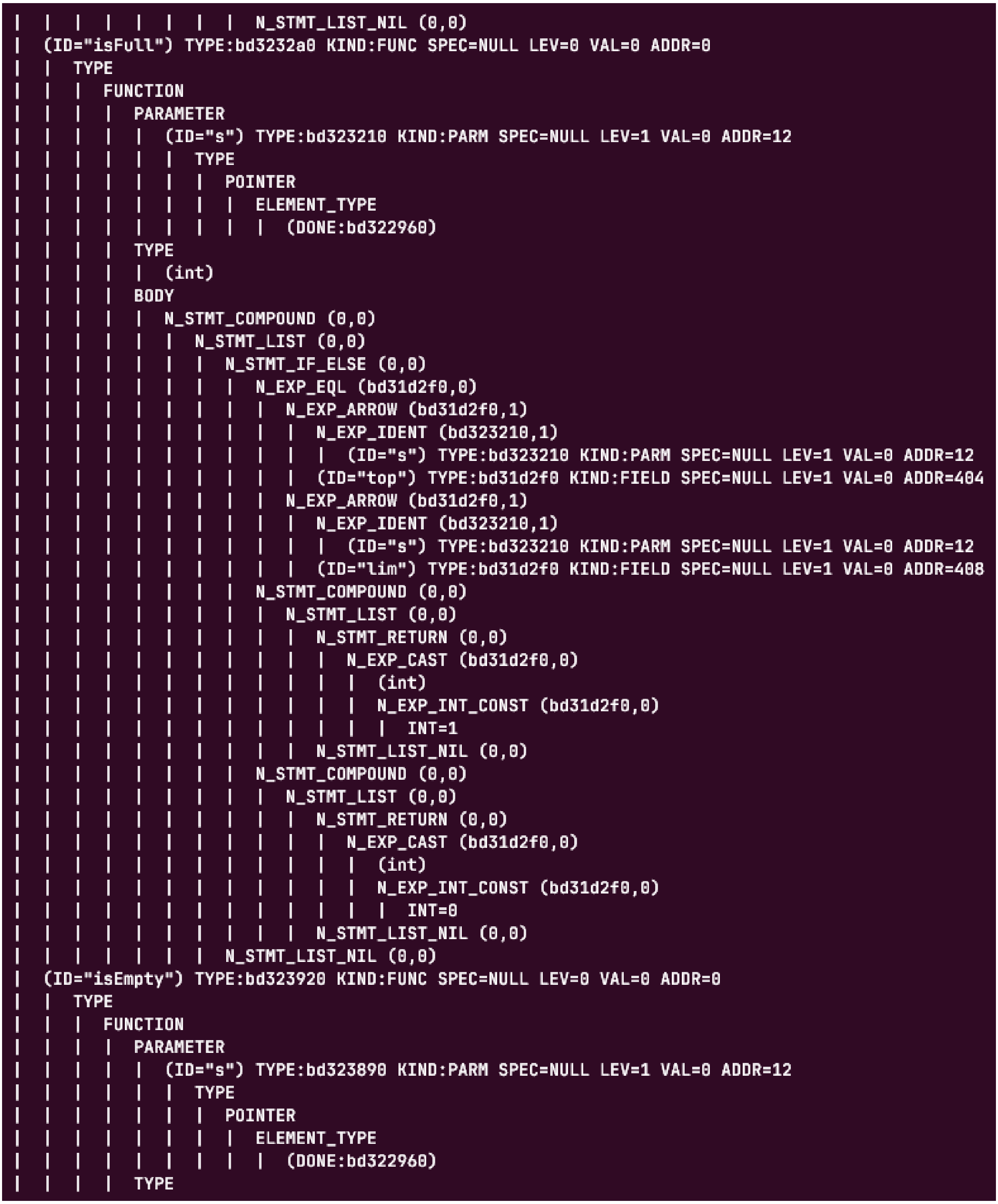


```
kh@ThinkPad-T16g2:~/compiler/hw6/cmake-build-debug$ ./a.out < ../testdir/8.c
=======  syntax tree  ==========
N_PROGRAM (0,0)
|  (ID="Stack") TYPE:bd322960 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  STRUCT
|  |  |  |  FIELD
|  |  |  |  |  (ID="arr") TYPE:bd322a50 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  ARRAY
|  |  |  |  |  |  |  |  INDEX
|  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (0,0)
```

(syntax tree 생략)

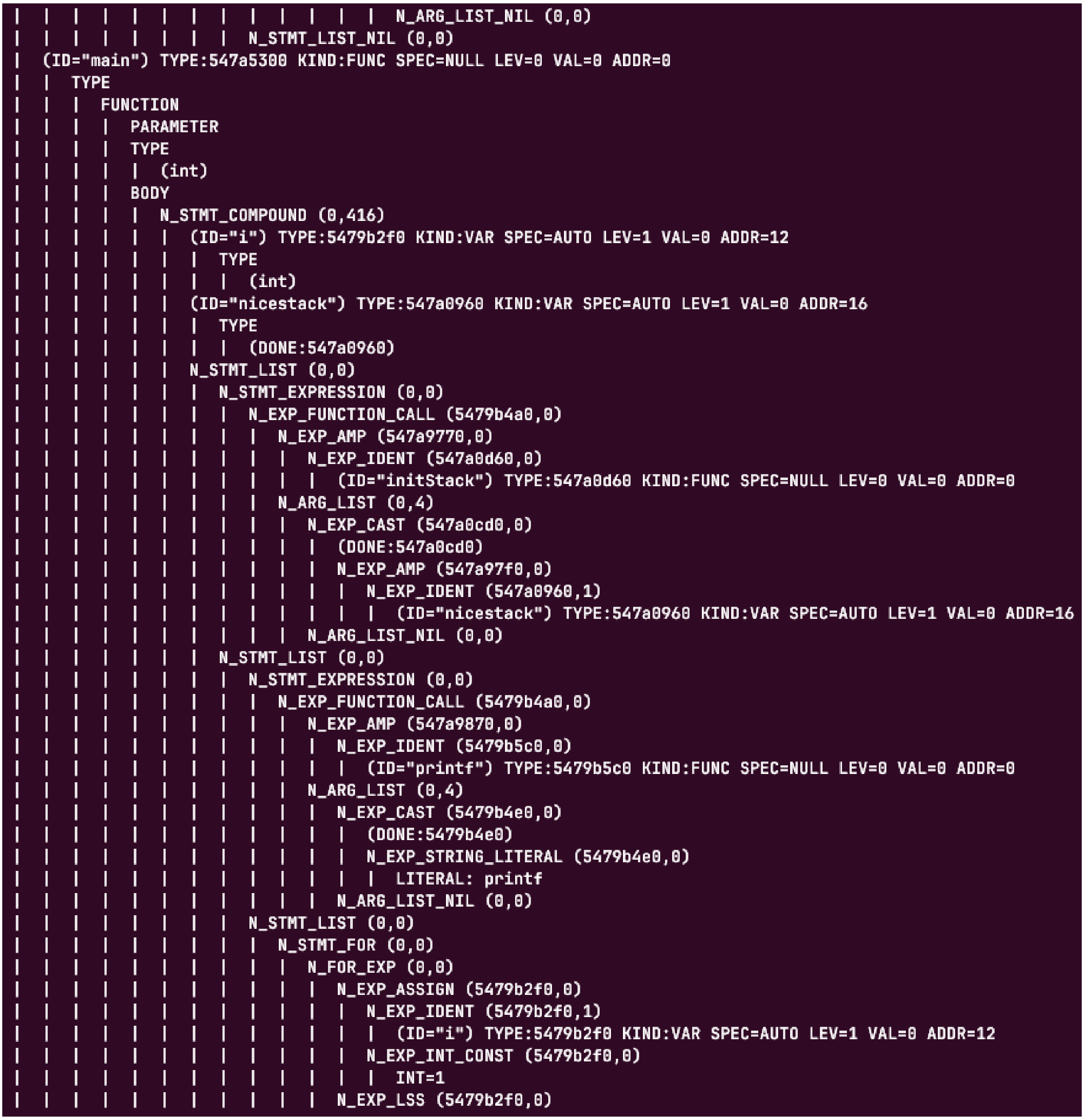


```
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
=======  semantic tree  ==========
N_PROGRAM (0,132)
|  (ID="Stack") TYPE:bd322960 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  STRUCT
|  |  |  |  FIELD
|  |  |  |  |  (ID="arr") TYPE:bd322a50 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  ARRAY
|  |  |  |  |  |  |  |  INDEX
|  |  |  |  |  |  |  |  |  INT=101
|  |  |  |  |  |  |  |  ELEMENT_TYPE
|  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  (ID="top") TYPE:bd31d2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=404
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (int)
|  |  |  |  |  (ID="lim") TYPE:bd31d2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=408
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  (int)
|  (ID="initStack") TYPE:bd322d60 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  |  (ID="s") TYPE:bd322cd0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  POINTER
|  |  |  |  |  |  |  |  ELEMENT_TYPE
|  |  |  |  |  |  |  |  |  (DONE:bd322960)
|  |  |  |  TYPE
|  |  |  |  |  (void)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  N_EXP_ASSIGN (bd31d2f0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ARROW (bd31d2f0,1)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (bd322cd0,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="s") TYPE:bd322cd0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  (ID="top") TYPE:bd31d2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=404
|  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (bd31d2f0,0)
|  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (bd31d2f0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_ARROW (bd31d2f0,1)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (bd322cd0,1)
|  |  |  |  |  |  |  |  |  |  |  |  (ID="s") TYPE:bd322cd0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  |  (ID="lim") TYPE:bd31d2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=408
|  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (bd31d2f0,0)
|  |  |  |  |  |  |  |  |  |  |  INT=100
```



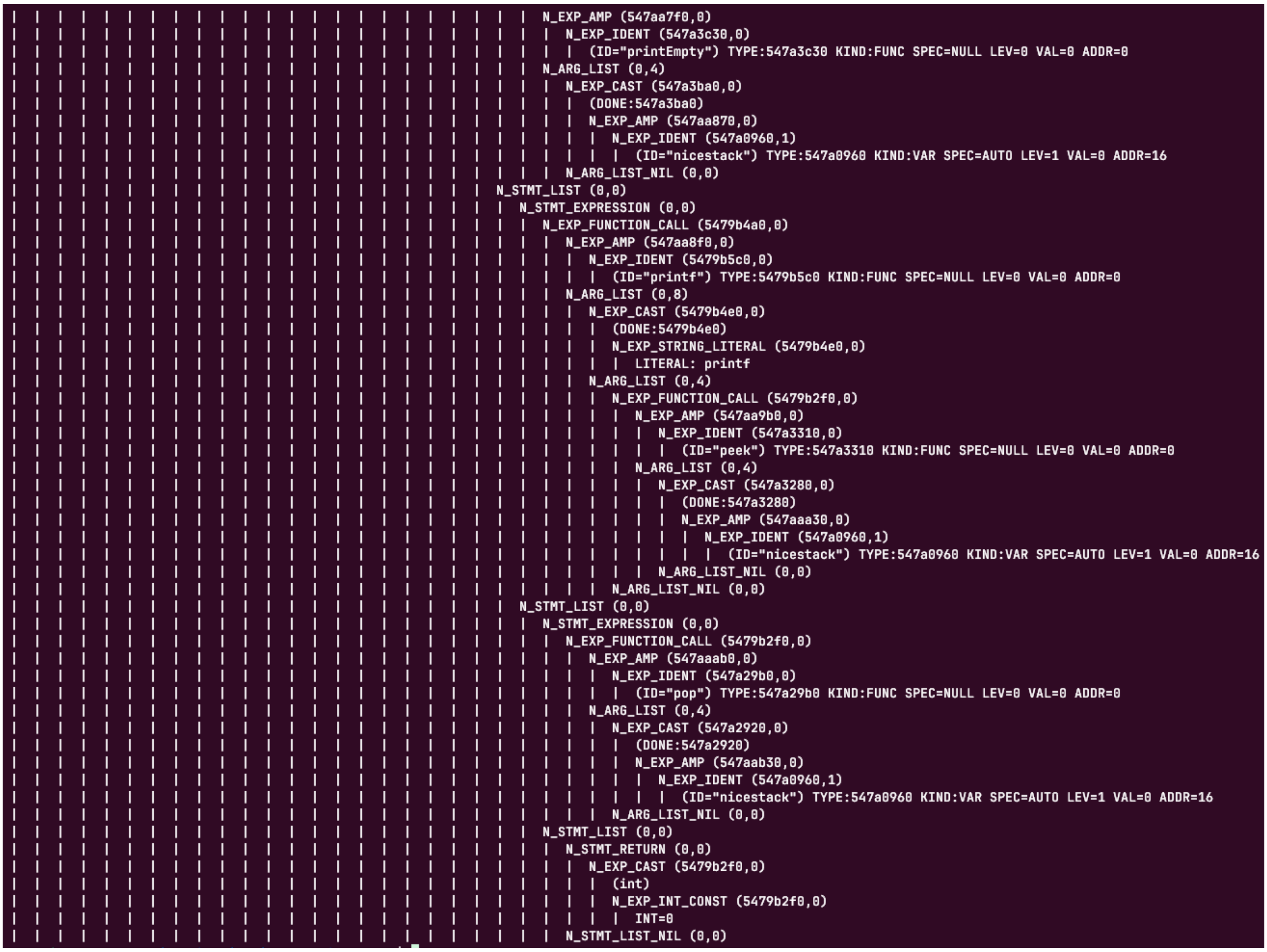
```
|  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  (ID="isFull") TYPE:bd3232a0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  |  (ID="s") TYPE:bd323210 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  POINTER
|  |  |  |  |  |  |  |  ELEMENT_TYPE
|  |  |  |  |  |  |  |  |  (DONE:bd322960)
|  |  |  |  TYPE
|  |  |  |  |  (int)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_IF_ELSE (0,0)
|  |  |  |  |  |  |  |  N_EXP_EQL (bd31d2f0,0)
|  |  |  |  |  |  |  |  |  N_EXP_ARROW (bd31d2f0,1)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (bd323210,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="s") TYPE:bd323210 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  (ID="top") TYPE:bd31d2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=404
|  |  |  |  |  |  |  |  |  N_EXP_ARROW (bd31d2f0,1)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (bd323210,1)
|  |  |  |  |  |  |  |  |  |  |  (ID="s") TYPE:bd323210 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  (ID="lim") TYPE:bd31d2f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=408
|  |  |  |  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_RETURN (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (bd31d2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (bd31d2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  N_STMT_COMPOUND (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  N_STMT_RETURN (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (bd31d2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (bd31d2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  (ID="isEmpty") TYPE:bd323920 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  |  (ID="s") TYPE:bd323890 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  POINTER
|  |  |  |  |  |  |  |  ELEMENT_TYPE
|  |  |  |  |  |  |  |  |  (DONE:bd322960)
|  |  |  |  TYPE
```

(semantic tree 일부 생략)



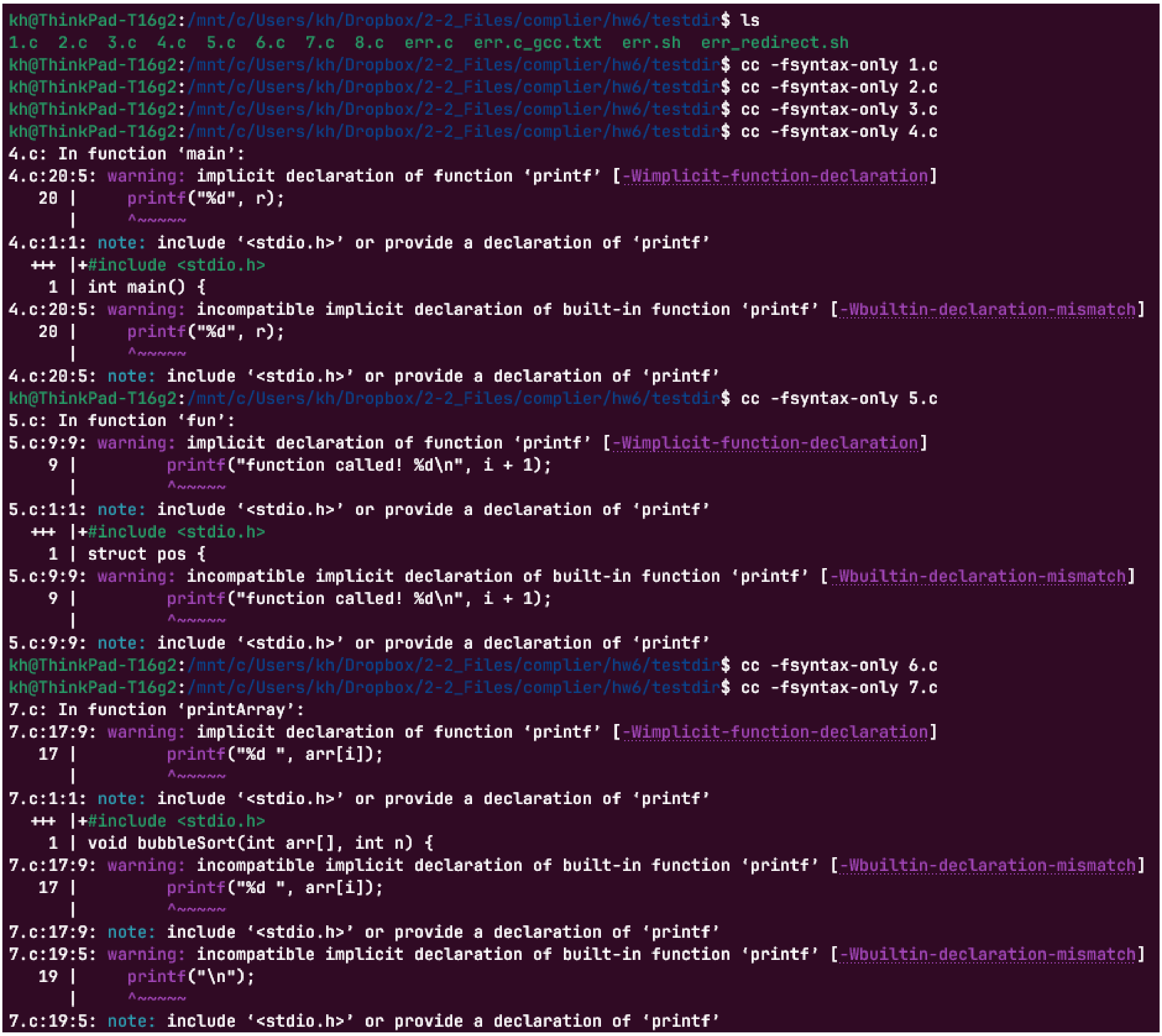
```
|  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
|  (ID="main") TYPE:547a5300 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  TYPE
|  |  |  FUNCTION
|  |  |  |  PARAMETER
|  |  |  |  TYPE
|  |  |  |  |  (int)
|  |  |  |  BODY
|  |  |  |  |  N_STMT_COMPOUND (0,416)
|  |  |  |  |  |  (ID="i") TYPE:5479b2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  (ID="nicestack") TYPE:547a0960 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  TYPE
|  |  |  |  |  |  |  |  (DONE:547a0960)
|  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (5479b4a0,0)
|  |  |  |  |  |  |  |  |  N_EXP_AMP (547a9770,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (547a0d60,0)
|  |  |  |  |  |  |  |  |  |  |  (ID="initStack") TYPE:547a0d60 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  N_EXP_CAST (547a0cd0,0)
|  |  |  |  |  |  |  |  |  |  |  (DONE:547a0cd0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (547a97f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (547a0960,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  (ID="nicestack") TYPE:547a0960 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (5479b4a0,0)
|  |  |  |  |  |  |  |  |  |  N_EXP_AMP (547a9870,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (5479b5c0,0)
|  |  |  |  |  |  |  |  |  |  |  |  (ID="printf") TYPE:5479b5c0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (5479b4e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  (DONE:5479b4e0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRING_LITERAL (5479b4e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: printf
|  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  N_STMT_FOR (0,0)
|  |  |  |  |  |  |  |  |  |  N_FOR_EXP (0,0)
|  |  |  |  |  |  |  |  |  |  |  N_EXP_ASSIGN (5479b2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (5479b2f0,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  (ID="i") TYPE:5479b2f0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=12
|  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (5479b2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  INT=1
|  |  |  |  |  |  |  |  |  |  |  N_EXP_LSS (5479b2f0,0)
```

(semantic tree 일부 생략)

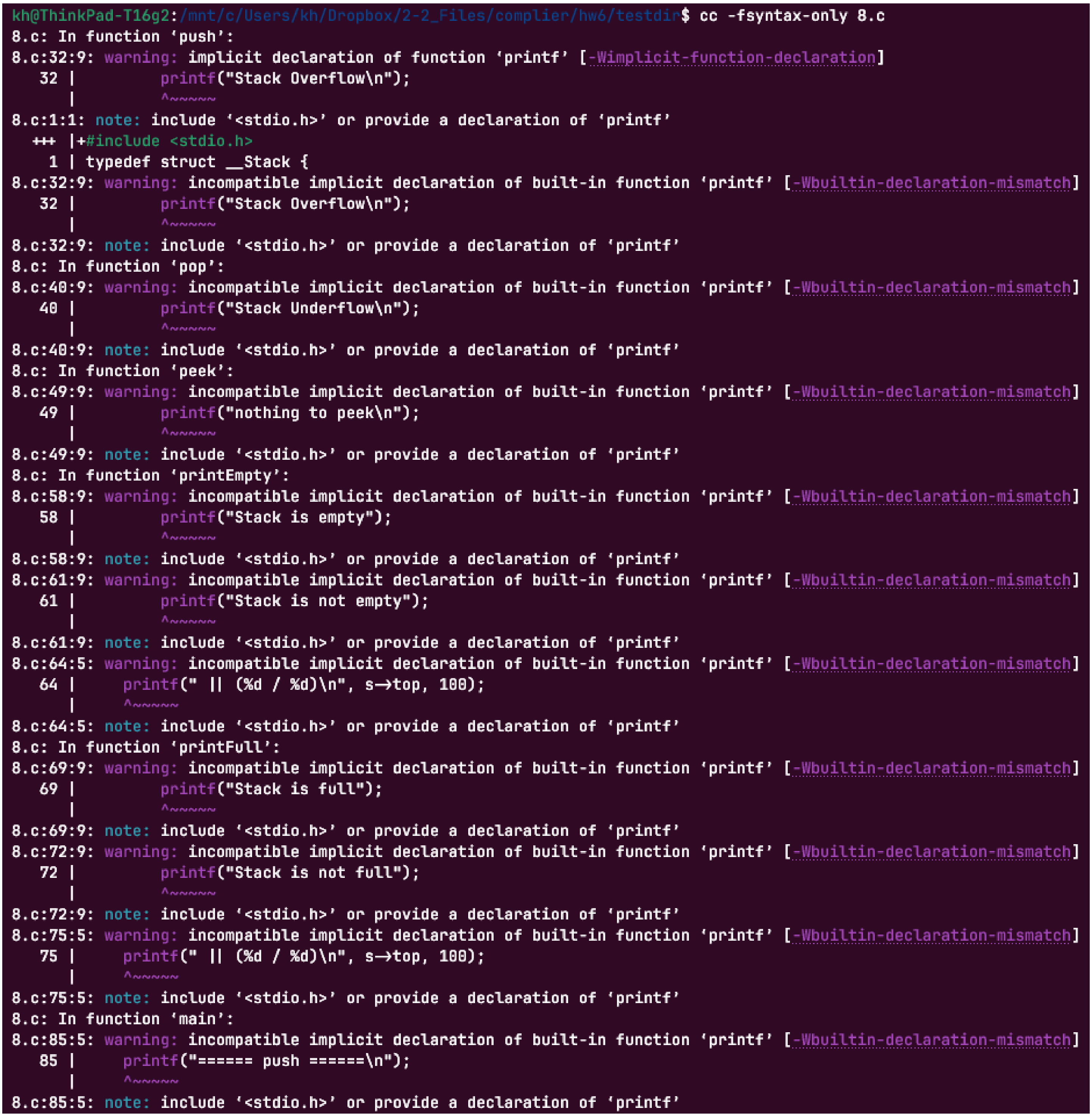


```
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (547aa7f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (547a3c30,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="printEmpty") TYPE:547a3c30 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (547a3ba0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (DONE:547a3ba0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (547aa870,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (547a0960,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="nicestack") TYPE:547a0960 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (5479b4a0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (547aa8f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (5479b5c0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="printf") TYPE:5479b5c0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,8)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (5479b4e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (DONE:5479b4e0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_STRING_LITERAL (5479b4e0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  LITERAL: printf
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (5479b2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (547aa9b0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (547a3310,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="peek") TYPE:547a3310 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (547a3280,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (DONE:547a3280)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (547aaa30,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (547a0960,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="nicestack") TYPE:547a0960 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_EXPRESSION (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_FUNCTION_CALL (5479b2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (547aaab0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (547a29b0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="pop") TYPE:547a29b0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST (0,4)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (547a2920,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (DONE:547a2920)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_AMP (547aab30,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_IDENT (547a0960,1)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (ID="nicestack") TYPE:547a0960 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=16
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_ARG_LIST_NIL (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_RETURN (0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_CAST (5479b2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  (int)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_EXP_INT_CONST (5479b2f0,0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  INT=0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  N_STMT_LIST_NIL (0,0)
```

3-9. gcc와 비교



```
kh@ThinkPad-T16g2:/mnt/c/Users/kh/Dropbox/2-2_Files/complier/hw6/testdir$ ls
1.c  2.c  3.c  4.c  5.c  6.c  7.c  8.c  err.c  err.c_gcc.txt  err.sh  err_redirect.sh
kh@ThinkPad-T16g2:/mnt/c/Users/kh/Dropbox/2-2_Files/complier/hw6/testdir$ cc -fsyntax-only 1.c
kh@ThinkPad-T16g2:/mnt/c/Users/kh/Dropbox/2-2_Files/complier/hw6/testdir$ cc -fsyntax-only 2.c
kh@ThinkPad-T16g2:/mnt/c/Users/kh/Dropbox/2-2_Files/complier/hw6/testdir$ cc -fsyntax-only 3.c
kh@ThinkPad-T16g2:/mnt/c/Users/kh/Dropbox/2-2_Files/complier/hw6/testdir$ cc -fsyntax-only 4.c
4.c: In function ‘main’:
4.c:20:5: warning: implicit declaration of function ‘printf’ [-Wimplicit-function-declaration]
   20 |     printf("%d", r);
      |     ^~~~~~
4.c:1:1: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
  +++ |+#include <stdio.h>
    1 | int main() {
4.c:20:5: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   20 |     printf("%d", r);
      |     ^~~~~~
4.c:20:5: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
kh@ThinkPad-T16g2:/mnt/c/Users/kh/Dropbox/2-2_Files/complier/hw6/testdir$ cc -fsyntax-only 5.c
5.c: In function ‘fun’:
5.c:9:9: warning: implicit declaration of function ‘printf’ [-Wimplicit-function-declaration]
    9 |         printf("function called! %d\n", i + 1);
      |         ^~~~~~
5.c:1:1: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
  +++ |+#include <stdio.h>
    1 | struct pos {
5.c:9:9: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
    9 |         printf("function called! %d\n", i + 1);
      |         ^~~~~~
5.c:9:9: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
kh@ThinkPad-T16g2:/mnt/c/Users/kh/Dropbox/2-2_Files/complier/hw6/testdir$ cc -fsyntax-only 6.c
kh@ThinkPad-T16g2:/mnt/c/Users/kh/Dropbox/2-2_Files/complier/hw6/testdir$ cc -fsyntax-only 7.c
7.c: In function ‘printArray’:
7.c:17:9: warning: implicit declaration of function ‘printf’ [-Wimplicit-function-declaration]
   17 |         printf("%d ", arr[i]);
      |         ^~~~~~
7.c:1:1: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
  +++ |+#include <stdio.h>
    1 | void bubbleSort(int arr[], int n) {
7.c:17:9: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   17 |         printf("%d ", arr[i]);
      |         ^~~~~~
7.c:17:9: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
7.c:19:5: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   19 |     printf("\n");
      |     ^~~~~~
7.c:19:5: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
```



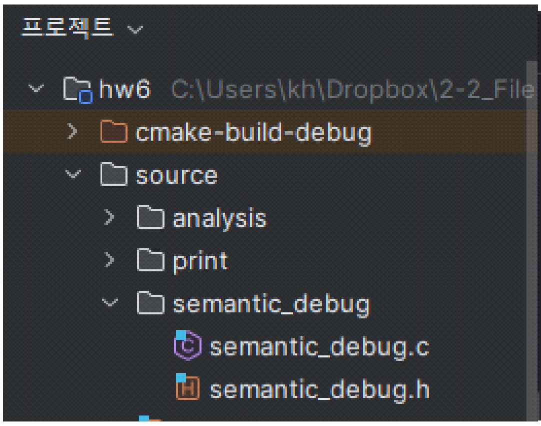
```
kh@ThinkPad-T16g2:/mnt/c/Users/kh/Dropbox/2-2_Files/complier/hw6/testdir$ cc -fsyntax-only 8.c
8.c: In function ‘push’:
8.c:32:9: warning: implicit declaration of function ‘printf’ [-Wimplicit-function-declaration]
   32 |         printf("Stack Overflow\n");
      |         ^~~~~~
8.c:1:1: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
  +++ |+#include <stdio.h>
    1 | typedef struct __Stack {
8.c:32:9: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   32 |         printf("Stack Overflow\n");
      |         ^~~~~~
8.c:32:9: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
8.c: In function ‘pop’:
8.c:40:9: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   40 |         printf("Stack Underflow\n");
      |         ^~~~~~
8.c:40:9: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
8.c: In function ‘peek’:
8.c:49:9: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   49 |         printf("nothing to peek\n");
      |         ^~~~~~
8.c:49:9: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
8.c: In function ‘printEmpty’:
8.c:58:9: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   58 |         printf("Stack is empty");
      |         ^~~~~~
8.c:58:9: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
8.c:61:9: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   61 |         printf("Stack is not empty");
      |         ^~~~~~
8.c:61:9: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
8.c:64:5: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   64 |     printf(" || (%d / %d)\n", s->top, 100);
      |     ^~~~~~
8.c:64:5: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
8.c: In function ‘printFull’:
8.c:69:9: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   69 |         printf("Stack is full");
      |         ^~~~~~
8.c:69:9: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
8.c:72:9: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   72 |         printf("Stack is not full");
      |         ^~~~~~
8.c:72:9: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
8.c:75:5: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   75 |     printf(" || (%d / %d)\n", s->top, 100);
      |     ^~~~~~
8.c:75:5: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
8.c: In function ‘main’:
8.c:85:5: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   85 |     printf("====== push ======\n");
      |     ^~~~~~
8.c:85:5: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
```

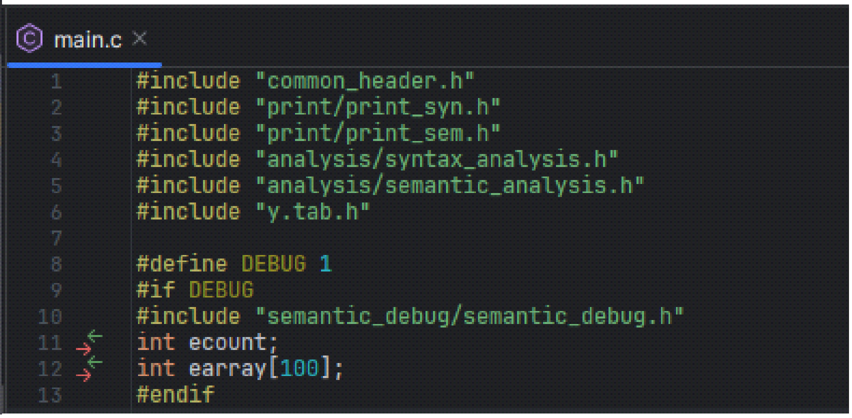
과제 프로그램은 #include를 파싱하지 못하기에 테스트 코드에 일부러 넣지 않은 stdio.h 헤더가 없어서 나오는 printf 경고 이외에는 딱히 발생하는 오류가 없었다.

4. 테스트 (2) - 의미적으로 올바르지 않은 코드

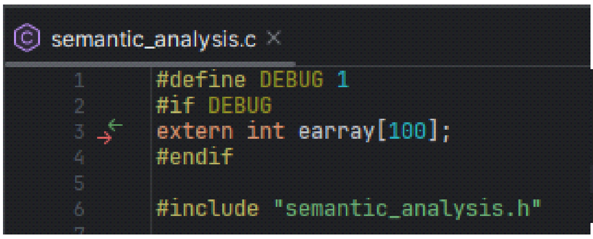
교재의 7장, semantic_error()을 살펴보며 최대한 많이 찾으려고 노력했습니다.

어떤 경우의 에러가 발생했는지 종합하기가 힘들어

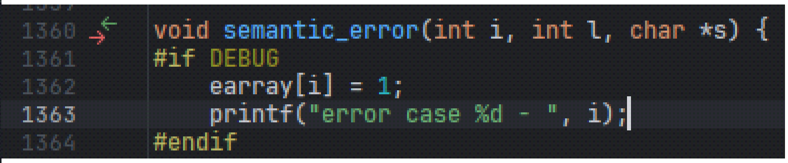




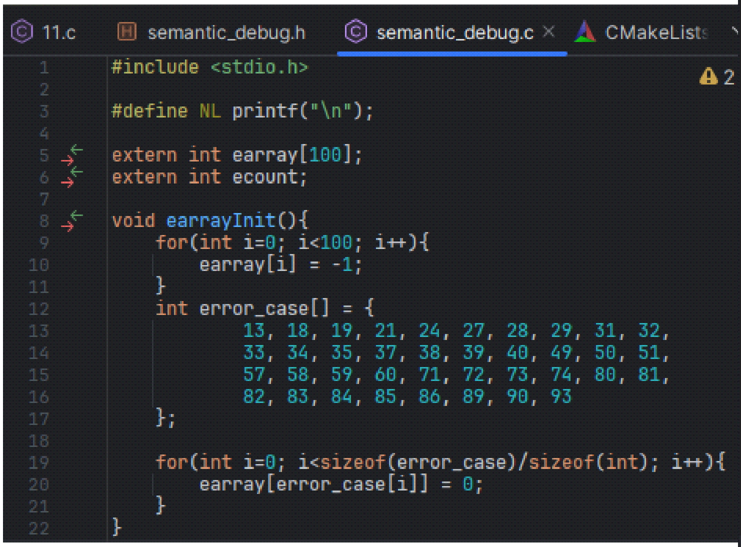
```c
#include "common_header.h"
#include "print/print_syn.h"
#include "print/print_sem.h"
#include "analysis/syntax_analysis.h"
#include "analysis/semantic_analysis.h"
#include "y.tab.h"

#define DEBUG 1
#if DEBUG
#include "semantic_debug/semantic_debug.h"
int ecount;
int earray[100];
#endif
```



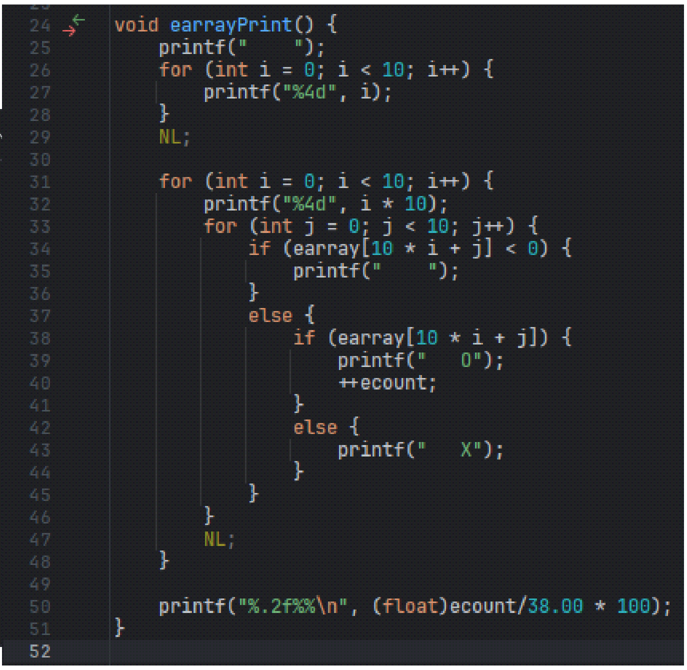
```c
#define DEBUG 1
#if DEBUG
extern int earray[100];
#endif

#include "semantic_analysis.h"
```



```c
void semantic_error(int i, int l, char *s) {
#if DEBUG
    earray[i] = 1;
    printf("error case %d - ", i);
#endif
```

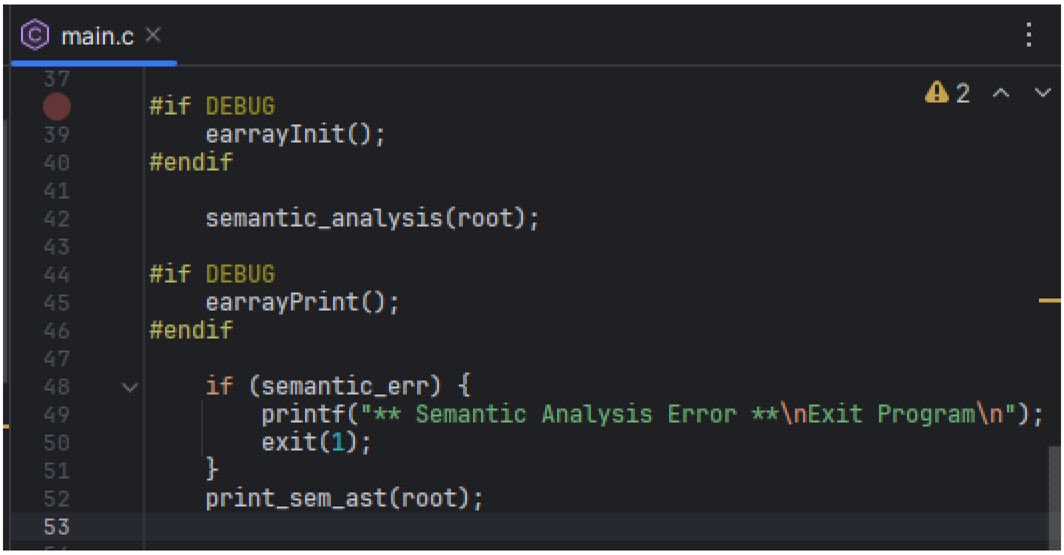


```c
#include <stdio.h>

#define NL printf("\n");

extern int earray[100];
extern int ecount;

void earrayInit(){
    for(int i=0; i<100; i++){
        earray[i] = -1;
    }
    int error_case[] = {
            13, 18, 19, 21, 24, 27, 28, 29, 31, 32,
            33, 34, 35, 37, 38, 39, 40, 49, 50, 51,
            57, 58, 59, 60, 71, 72, 73, 74, 80, 81,
            82, 83, 84, 85, 86, 89, 90, 93
    };

    for(int i=0; i<sizeof(error_case)/sizeof(int); i++){
        earray[error_case[i]] = 0;
    }
}
```

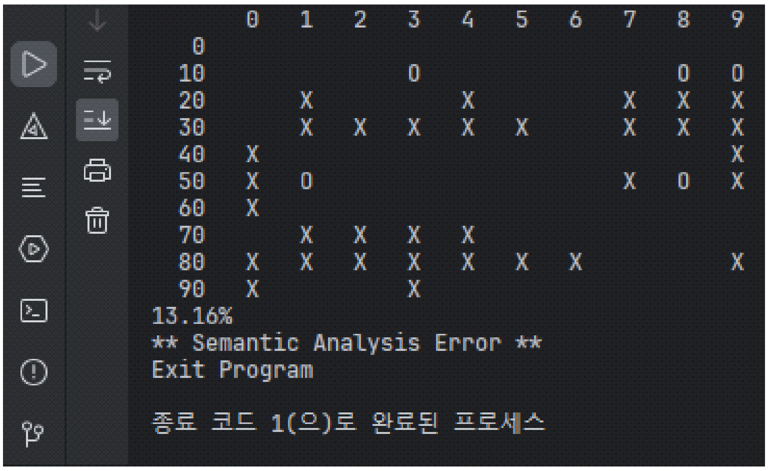


```c
void earrayPrint() {
    printf("    ");
    for (int i = 0; i < 10; i++) {
        printf("%4d", i);
    }
    NL;

    for (int i = 0; i < 10; i++) {
        printf("%4d", i * 10);
        for (int j = 0; j < 10; j++) {
            if (earray[10 * i + j] < 0) {
                printf("    ");
            }
            else {
                if (earray[10 * i + j]) {
                    printf("   O");
                    ++ecount;
                }
                else {
                    printf("   X");
                }
            }
        }
        NL;
    }

    printf("%.2f%%\n", (float)ecount/38.00 * 100);
}
```

프로그램에 위와 같이 추가하여



```c
#if DEBUG
    earrayInit();
#endif

    semantic_analysis(root);

#if DEBUG
    earrayPrint();
#endif

    if (semantic_err) {
        printf("** Semantic Analysis Error **\nExit Program\n");
        exit(1);
    }
    print_sem_ast(root);
```

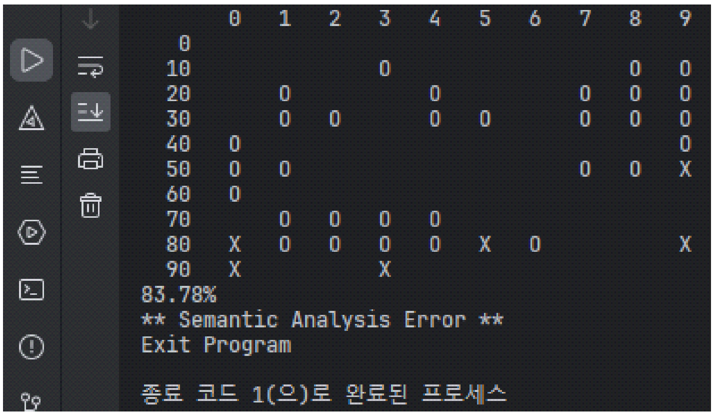


```
       0   1   2   3   4   5   6   7   8   9
   0
  10               O                   O   O
  20       X           X           X   X   X
  30       X   X   X   X   X       X   X   X
  40   X                                   X
  50   X   O                       X   O   X
  60   X
  70       X   X   X   X
  80   X   X   X   X   X   X   X           X
  90   X           X
13.16%
** Semantic Analysis Error **
Exit Program

종료 코드 1(으)로 완료된 프로세스
```

semantic tree의 출력전에 위와 같이 출력하게끔 프로그램을 수정했다. 요컨대 비어있지 않은 모든 곳을 O로 만들어야 한다.

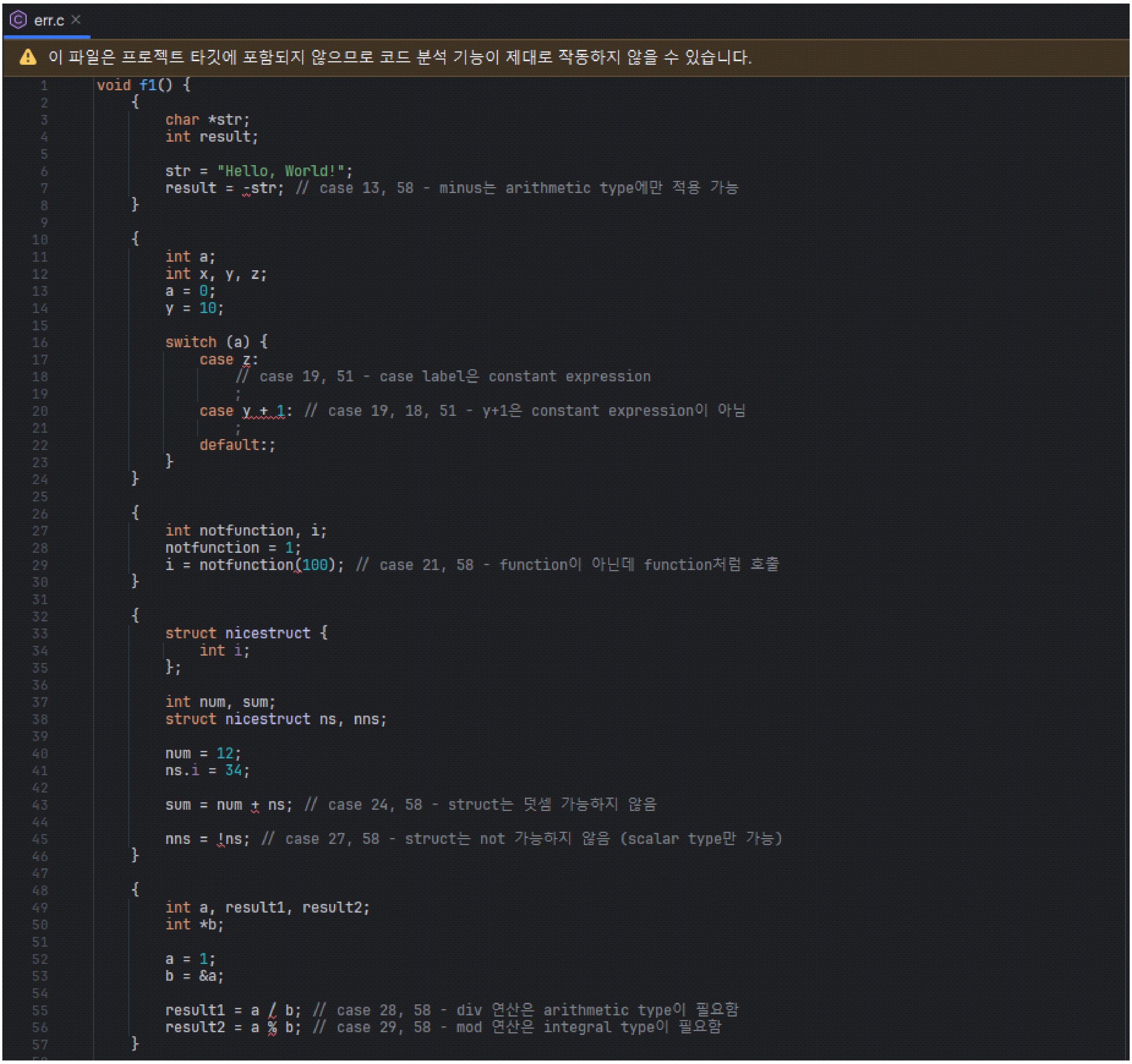
4-1. err.c

문법적으로는 맞는데 의미적으로 틀린 코드를 생각 해내는게 정말 힘들었고, 마지막 날 제출을 하는 이유이다.



```
       0   1   2   3   4   5   6   7   8   9
   0
  10               O                   O   O
  20       O           O           O   O   O
  30       O   O       O   O       O   O   O
  40   O                                   O
  50   O   O                       O   O   X
  60   O
  70       O   O   O   O
  80   X   O   O   O   O   X   O           X
  90   X           X
83.78%
** Semantic Analysis Error **
Exit Program

종료 코드 1(으)로 완료된 프로세스
```

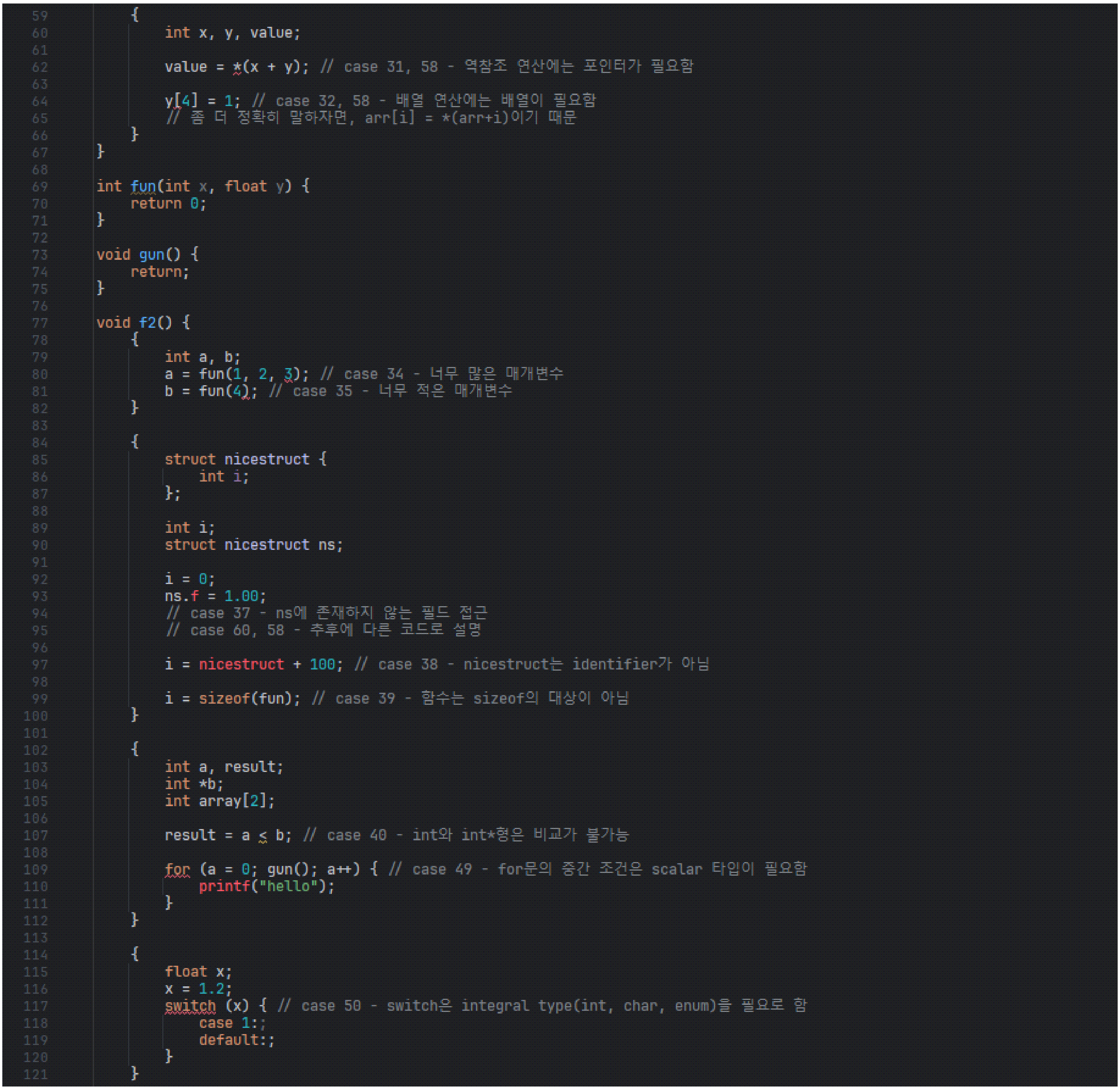
미리 결과부터 말하자면 semantic_error의 케이스 거의 모두에서 걸리는 테스트 코드를 만들었다. 만들어내지 못한 에러는 case 59, 80, 85, 89, 90, 93인데, 90과 93은 실질적으로 만들기가 불가능한 것을 생각해보면 4개를 만들지 못했다.



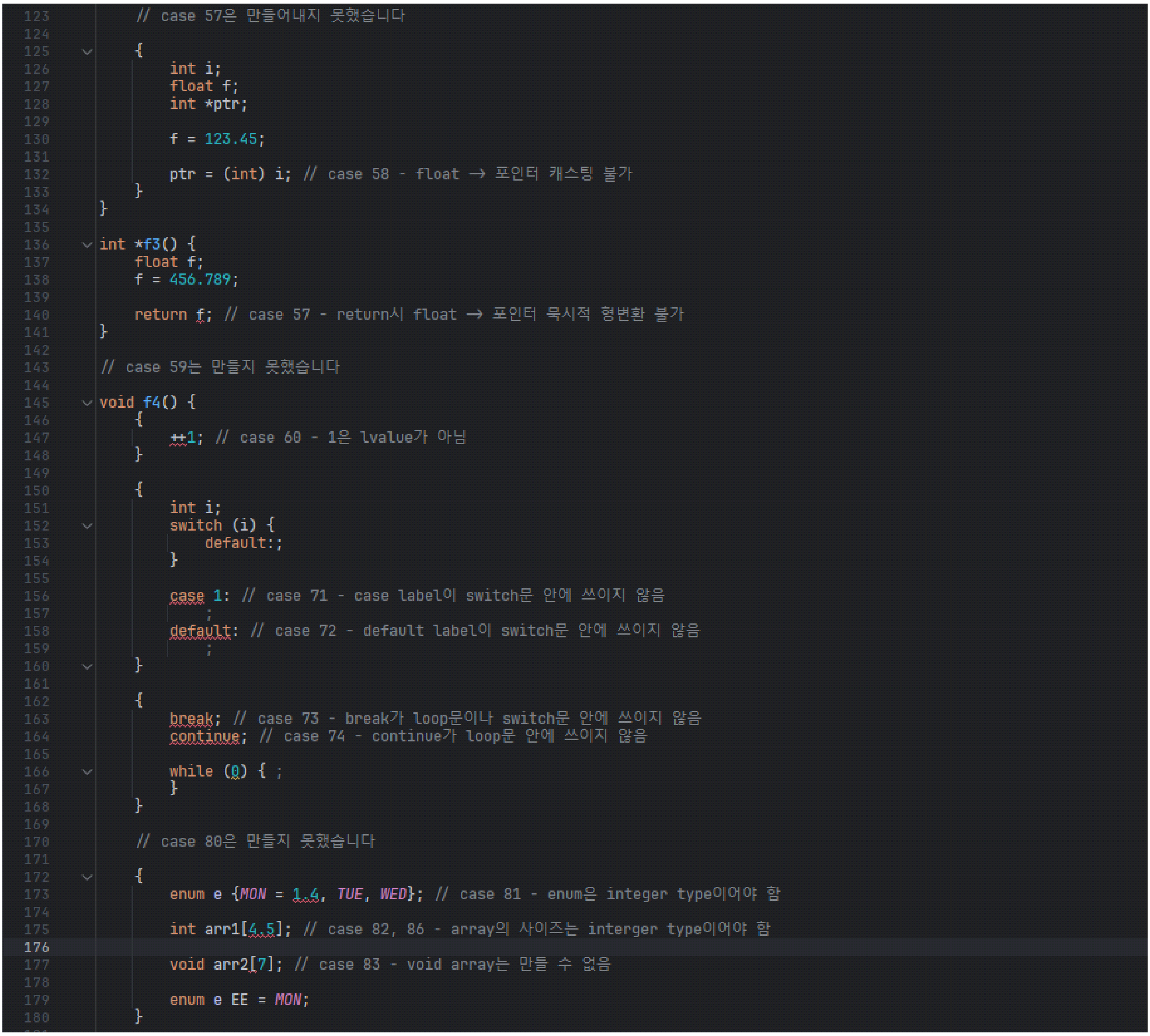
err.c ×

⚠ 이 파일은 프로젝트 타깃에 포함되지 않으므로 코드 분석 기능이 제대로 작동하지 않을 수 있습니다.

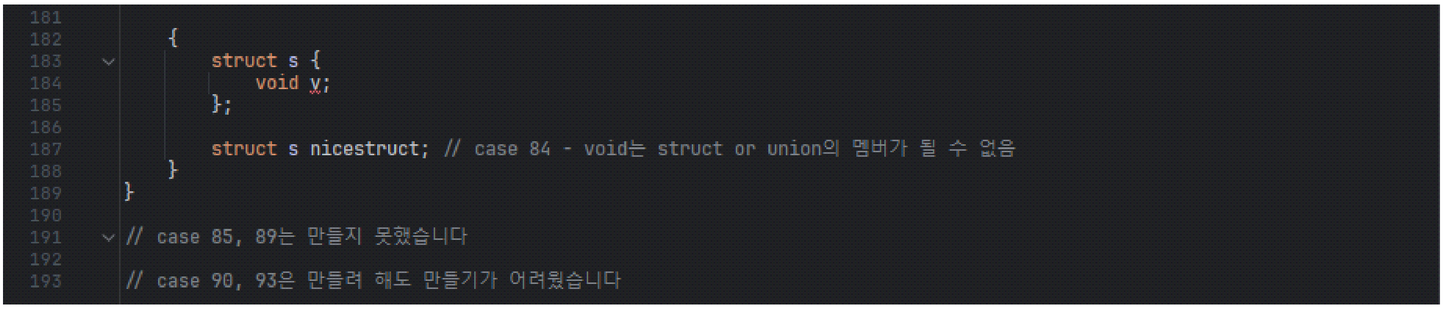
```c
void f1() {
    {
        char *str;
        int result;

        str = "Hello, World!";
        result = -str; // case 13, 58 - minus는 arithmetic type에만 적용 가능
    }

    {
        int a;
        int x, y, z;
        a = 0;
        y = 10;

        switch (a) {
            case z:
                // case 19, 51 - case label은 constant expression
                ;
            case y + 1: // case 19, 18, 51 - y+1은 constant expression이 아님
                ;
            default:;
        }
    }

    {
        int notfunction, i;
        notfunction = 1;
        i = notfunction(100); // case 21, 58 - function이 아닌데 function처럼 호출
    }

    {
        struct nicestruct {
            int i;
        };

        int num, sum;
        struct nicestruct ns, nns;

        num = 12;
        ns.i = 34;

        sum = num + ns; // case 24, 58 - struct는 덧셈 가능하지 않음

        nns = !ns; // case 27, 58 - struct는 not 가능하지 않음 (scalar type만 가능)
    }

    {
        int a, result1, result2;
        int *b;

        a = 1;
        b = &a;

        result1 = a / b; // case 28, 58 - div 연산은 arithmetic type이 필요함
        result2 = a % b; // case 29, 58 - mod 연산은 integral type이 필요함
    }
```



```c
    {
        int x, y, value;

        value = *(x + y); // case 31, 58 - 역참조 연산에는 포인터가 필요함

        y[4] = 1; // case 32, 58 - 배열 연산에는 배열이 필요함
        // 좀 더 정확히 말하자면, arr[i] = *(arr+i)이기 때문
    }
}

int fun(int x, float y) {
    return 0;
}

void gun() {
    return;
}

void f2() {
    {
        int a, b;
        a = fun(1, 2, 3); // case 34 - 너무 많은 매개변수
        b = fun(4); // case 35 - 너무 적은 매개변수
    }

    {
        struct nicestruct {
            int i;
        };

        int i;
        struct nicestruct ns;

        i = 0;
        ns.f = 1.00;
        // case 37 - ns에 존재하지 않는 필드 접근
        // case 60, 58 - 추후에 다른 코드로 설명

        i = nicestruct + 100; // case 38 - nicestruct는 identifier가 아님

        i = sizeof(fun); // case 39 - 함수는 sizeof의 대상이 아님
    }

    {
        int a, result;
        int *b;
        int array[2];

        result = a <= b; // case 40 - int와 int*형은 비교가 불가능

        for (a = 0; gun(); a++) { // case 49 - for문의 중간 조건은 scalar 타입이 필요함
            printf("hello");
        }
    }

    {
        float x;
        x = 1.2;
        switch (x) { // case 50 - switch은 integral type(int, char, enum)을 필요로 함
            case 1:;
            default:;
        }
    }
```



```c
    // case 57은 만들어내지 못했습니다

    {
        int i;
        float f;
        int *ptr;

        f = 123.45;

        ptr = (int) i; // case 58 - float → 포인터 캐스팅 불가
    }
}

int *f3() {
    float f;
    f = 456.789;

    return f; // case 57 - return시 float → 포인터 묵시적 형변환 불가
}

// case 59는 만들지 못했습니다

void f4() {
    {
        ++1; // case 60 - 1은 lvalue가 아님
    }

    {
        int i;
        switch (i) {
            default:;
        }

        case 1: // case 71 - case label이 switch문 안에 쓰이지 않음
            ;
        default: // case 72 - default label이 switch문 안에 쓰이지 않음
            ;
    }

    {
        break; // case 73 - break가 loop문이나 switch문 안에 쓰이지 않음
        continue; // case 74 - continue가 loop문 안에 쓰이지 않음

        while (0) { ;
        }
    }

    // case 80은 만들지 못했습니다

    {
        enum e {MON = 1.4, TUE, WED}; // case 81 - enum은 integer type이어야 함

        int arr1[4.5]; // case 82, 86 - array의 사이즈는 interger type이어야 함

        void arr2[7]; // case 83 - void array는 만들 수 없음

        enum e EE = MON;
    }
```

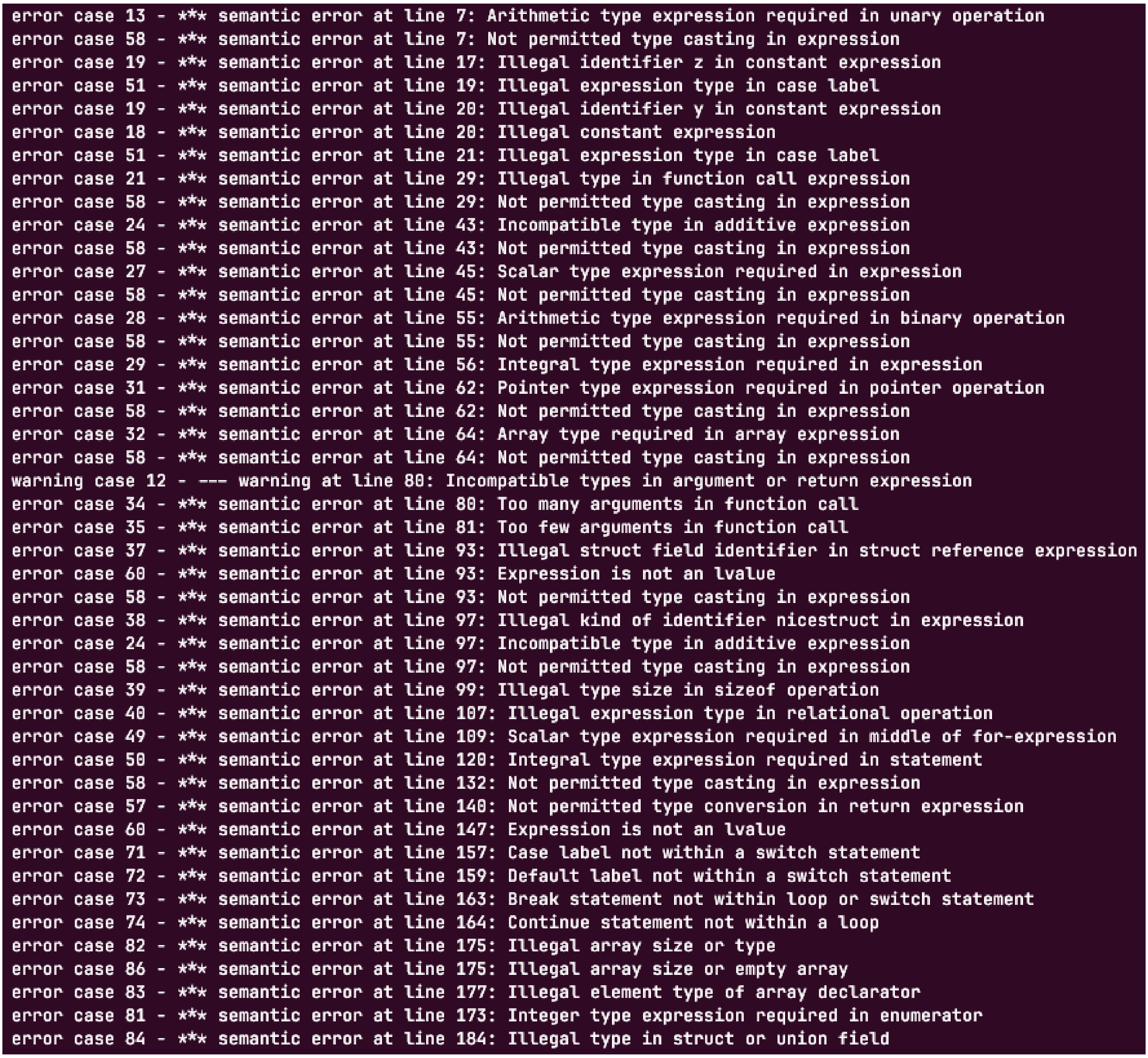


```c

    {
        struct s {
            void v;
        };

        struct s nicestruct; // case 84 - void는 struct or union의 멤버가 될 수 없음
    }
}

// case 85, 89는 만들지 못했습니다

// case 90, 93은 만들려 해도 만들기가 어려웠습니다
```

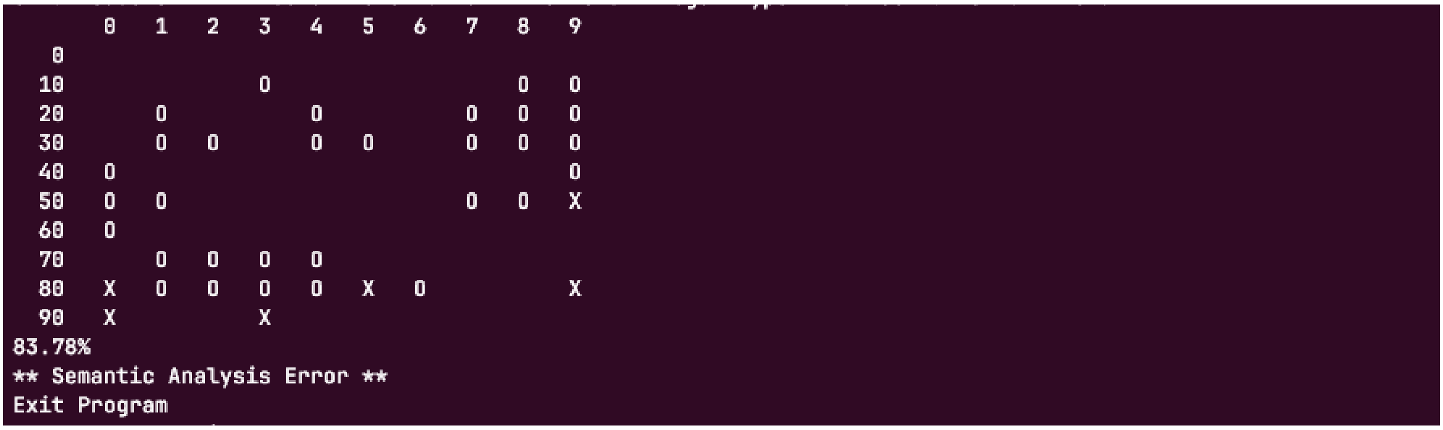
4-2. gcc와 비교

과제 프로그램은 다음과 같이 출력한다.

(syntax tree 출력 생략)



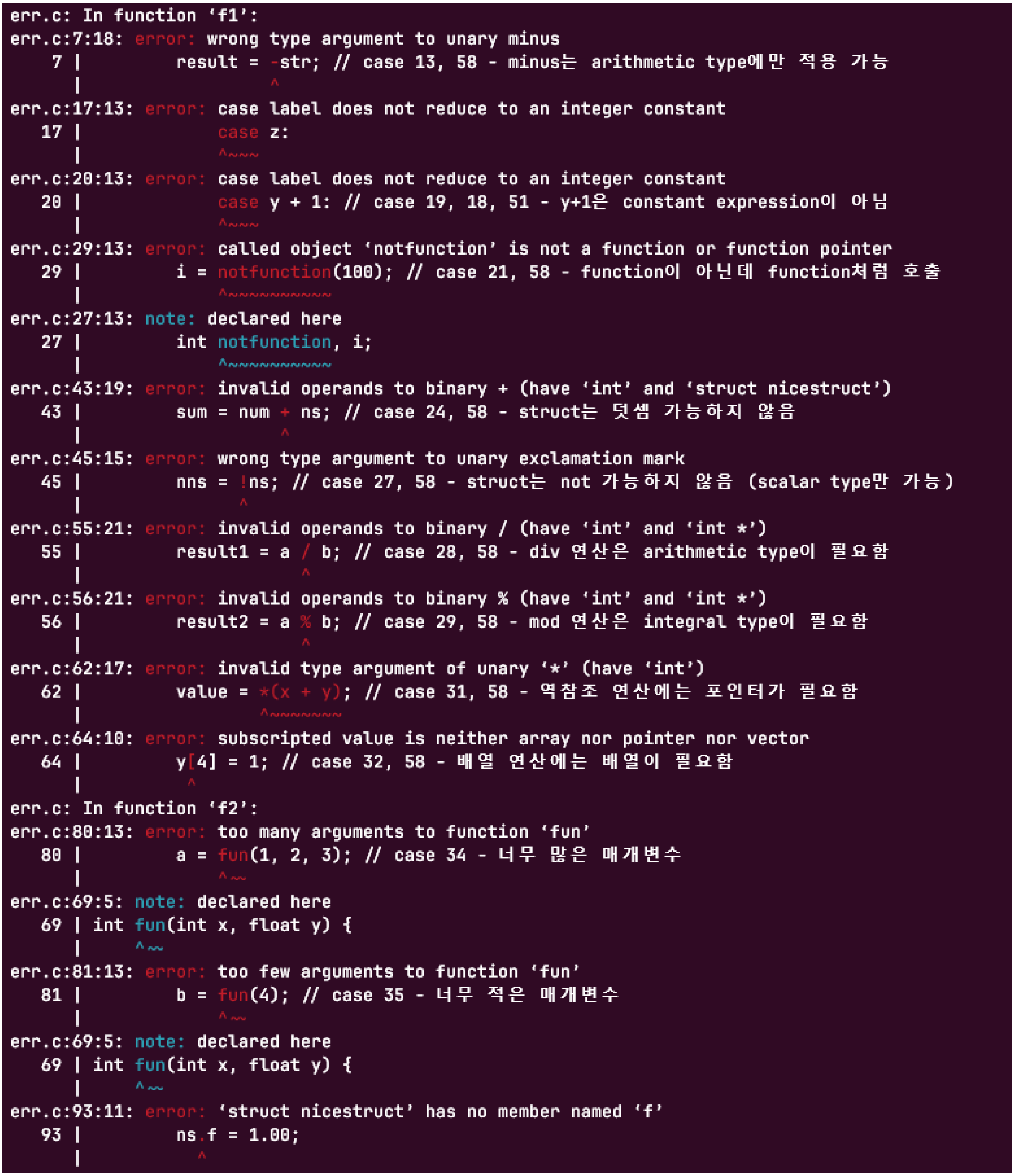
```
error case 13 - *** semantic error at line 7: Arithmetic type expression required in unary operation
error case 58 - *** semantic error at line 7: Not permitted type casting in expression
error case 19 - *** semantic error at line 17: Illegal identifier z in constant expression
error case 51 - *** semantic error at line 19: Illegal expression type in case label
error case 19 - *** semantic error at line 20: Illegal identifier y in constant expression
error case 18 - *** semantic error at line 20: Illegal constant expression
error case 51 - *** semantic error at line 21: Illegal expression type in case label
error case 21 - *** semantic error at line 29: Illegal type in function call expression
error case 58 - *** semantic error at line 29: Not permitted type casting in expression
error case 24 - *** semantic error at line 43: Incompatible type in additive expression
error case 58 - *** semantic error at line 43: Not permitted type casting in expression
error case 27 - *** semantic error at line 45: Scalar type expression required in expression
error case 58 - *** semantic error at line 45: Not permitted type casting in expression
error case 28 - *** semantic error at line 55: Arithmetic type expression required in binary operation
error case 58 - *** semantic error at line 55: Not permitted type casting in expression
error case 29 - *** semantic error at line 56: Integral type expression required in expression
error case 31 - *** semantic error at line 62: Pointer type expression required in pointer operation
error case 58 - *** semantic error at line 62: Not permitted type casting in expression
error case 32 - *** semantic error at line 64: Array type required in array expression
error case 58 - *** semantic error at line 64: Not permitted type casting in expression
warning case 12 - --- warning at line 80: Incompatible types in argument or return expression
error case 34 - *** semantic error at line 80: Too many arguments in function call
error case 35 - *** semantic error at line 81: Too few arguments in function call
error case 37 - *** semantic error at line 93: Illegal struct field identifier in struct reference expression
error case 60 - *** semantic error at line 93: Expression is not an lvalue
error case 58 - *** semantic error at line 93: Not permitted type casting in expression
error case 38 - *** semantic error at line 97: Illegal kind of identifier nicestruct in expression
error case 24 - *** semantic error at line 97: Incompatible type in additive expression
error case 58 - *** semantic error at line 97: Not permitted type casting in expression
error case 39 - *** semantic error at line 99: Illegal type size in sizeof operation
error case 40 - *** semantic error at line 107: Illegal expression type in relational operation
error case 49 - *** semantic error at line 109: Scalar type expression required in middle of for-expression
error case 50 - *** semantic error at line 120: Integral type expression required in statement
error case 58 - *** semantic error at line 132: Not permitted type casting in expression
error case 57 - *** semantic error at line 140: Not permitted type conversion in return expression
error case 60 - *** semantic error at line 147: Expression is not an lvalue
error case 71 - *** semantic error at line 157: Case label not within a switch statement
error case 72 - *** semantic error at line 159: Default label not within a switch statement
error case 73 - *** semantic error at line 163: Break statement not within loop or switch statement
error case 74 - *** semantic error at line 164: Continue statement not within a loop
error case 82 - *** semantic error at line 175: Illegal array size or type
error case 86 - *** semantic error at line 175: Illegal array size or empty array
error case 83 - *** semantic error at line 177: Illegal element type of array declarator
error case 81 - *** semantic error at line 173: Integer type expression required in enumerator
error case 84 - *** semantic error at line 184: Illegal type in struct or union field
```



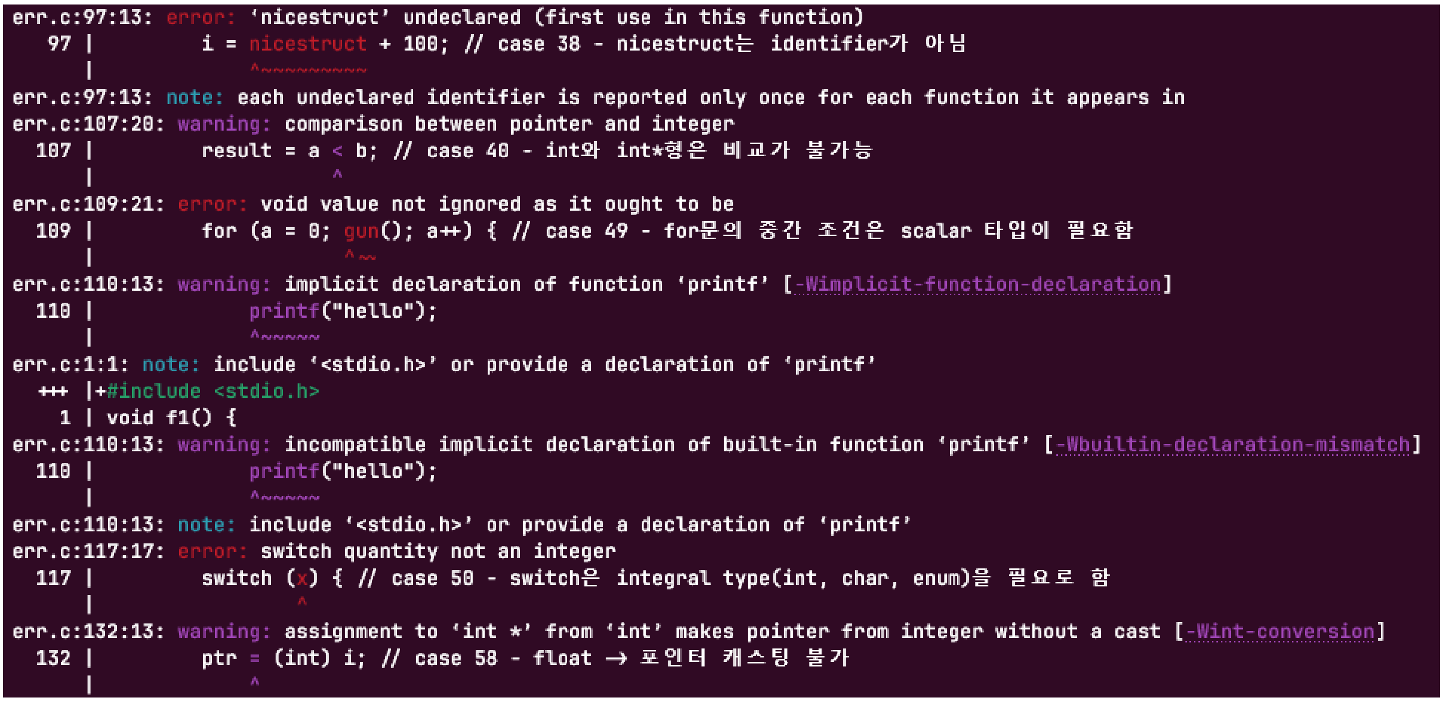
```
       0   1   2   3   4   5   6   7   8   9
   0
  10               O                   O   O
  20       O           O           O   O   O
  30       O   O       O   O       O   O   O
  40   O                                   O
  50   O   O                       O   O   X
  60   O
  70       O   O   O   O
  80   X   O   O   O   O   X   O           X
  90   X           X
83.78%
** Semantic Analysis Error **
Exit Program
```

(semantic_error 있기에 semantic tree 출력하지 않고 종료)

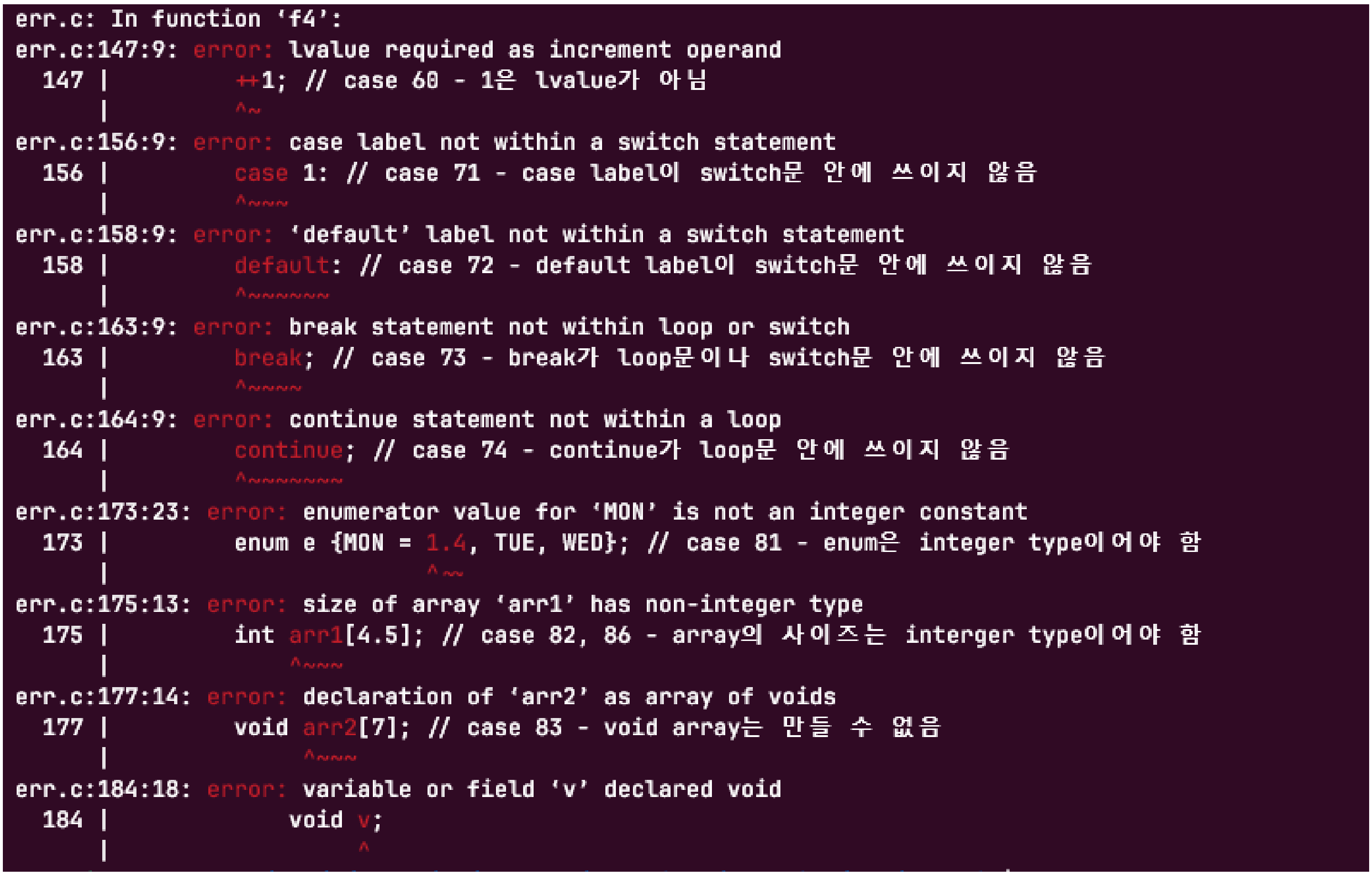
gcc는 다음과 같이 출력한다.



```
err.c: In function ‘f1’:
err.c:7:18: error: wrong type argument to unary minus
    7 |         result = -str; // case 13, 58 - minus는 arithmetic type에만 적용 가능
      |                  ^
err.c:17:13: error: case label does not reduce to an integer constant
   17 |             case z:
      |             ^~~~
err.c:20:13: error: case label does not reduce to an integer constant
   20 |             case y + 1: // case 19, 18, 51 - y+1은 constant expression이 아님
      |             ^~~~
err.c:29:13: error: called object ‘notfunction’ is not a function or function pointer
   29 |         i = notfunction(100); // case 21, 58 - function이 아닌데 function처럼 호출
      |             ^~~~~~~~~~~
err.c:27:13: note: declared here
   27 |         int notfunction, i;
      |             ^~~~~~~~~~~
err.c:43:19: error: invalid operands to binary + (have ‘int’ and ‘struct nicestruct’)
   43 |         sum = num + ns; // case 24, 58 - struct는 덧셈 가능하지 않음
      |                   ^
err.c:45:15: error: wrong type argument to unary exclamation mark
   45 |         nns = !ns; // case 27, 58 - struct는 not 가능하지 않음 (scalar type만 가능)
      |               ^
err.c:55:21: error: invalid operands to binary / (have ‘int’ and ‘int *’)
   55 |         result1 = a / b; // case 28, 58 - div 연산은 arithmetic type이 필요함
      |                     ^
err.c:56:21: error: invalid operands to binary % (have ‘int’ and ‘int *’)
   56 |         result2 = a % b; // case 29, 58 - mod 연산은 integral type이 필요함
      |                     ^
err.c:62:17: error: invalid type argument of unary ‘*’ (have ‘int’)
   62 |         value = *(x + y); // case 31, 58 - 역참조 연산에는 포인터가 필요함
      |                 ^~~~~~~~
err.c:64:10: error: subscripted value is neither array nor pointer nor vector
   64 |         y[4] = 1; // case 32, 58 - 배열 연산에는 배열이 필요함
      |          ^
err.c: In function ‘f2’:
err.c:80:13: error: too many arguments to function ‘fun’
   80 |         a = fun(1, 2, 3); // case 34 - 너무 많은 매개변수
      |             ^~~
err.c:69:5: note: declared here
   69 | int fun(int x, float y) {
      |     ^~~
err.c:81:13: error: too few arguments to function ‘fun’
   81 |         b = fun(4); // case 35 - 너무 적은 매개변수
      |             ^~~
err.c:69:5: note: declared here
   69 | int fun(int x, float y) {
      |     ^~~
err.c:93:11: error: ‘struct nicestruct’ has no member named ‘f’
   93 |         ns.f = 1.00;
      |           ^
```



```
err.c:97:13: error: ‘nicestruct’ undeclared (first use in this function)
   97 |         i = nicestruct + 100; // case 38 - nicestruct는 identifier가 아님
      |             ^~~~~~~~~~
err.c:97:13: note: each undeclared identifier is reported only once for each function it appears in
err.c:107:20: warning: comparison between pointer and integer
  107 |         result = a < b; // case 40 - int와 int*형은 비교가 불가능
      |                    ^
err.c:109:21: error: void value not ignored as it ought to be
  109 |         for (a = 0; gun(); a++) { // case 49 - for문의 중간 조건은 scalar 타입이 필요함
      |                     ^~~
err.c:110:13: warning: implicit declaration of function ‘printf’ [-Wimplicit-function-declaration]
  110 |             printf("hello");
      |             ^~~~~~
err.c:1:1: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
  +++ |+#include <stdio.h>
    1 | void f1() {
err.c:110:13: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
  110 |             printf("hello");
      |             ^~~~~~
err.c:110:13: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
err.c:117:17: error: switch quantity not an integer
  117 |         switch (x) { // case 50 - switch은 integral type(int, char, enum)을 필요로 함
      |                 ^
err.c:132:13: warning: assignment to ‘int *’ from ‘int’ makes pointer from integer without a cast [-Wint-conversion]
  132 |         ptr = (int) i; // case 58 - float → 포인터 캐스팅 불가
      |             ^
```



```
err.c: In function ‘f4’:
err.c:147:9: error: lvalue required as increment operand
  147 |         ++1; // case 60 - 1은 lvalue가 아님
      |         ^~
err.c:156:9: error: case label not within a switch statement
  156 |         case 1: // case 71 - case label이 switch문 안에 쓰이지 않음
      |         ^~~~
err.c:158:9: error: ‘default’ label not within a switch statement
  158 |         default: // case 72 - default label이 switch문 안에 쓰이지 않음
      |         ^~~~~~~
err.c:163:9: error: break statement not within loop or switch
  163 |         break; // case 73 - break가 loop문이나 switch문 안에 쓰이지 않음
      |         ^~~~~
err.c:164:9: error: continue statement not within a loop
  164 |         continue; // case 74 - continue가 loop문 안에 쓰이지 않음
      |         ^~~~~~~~
err.c:173:23: error: enumerator value for ‘MON’ is not an integer constant
  173 |         enum e {MON = 1.4, TUE, WED}; // case 81 - enum은 integer type이어야 함
      |                       ^~~
err.c:175:13: error: size of array ‘arr1’ has non-integer type
  175 |         int arr1[4.5]; // case 82, 86 - array의 사이즈는 interger type이어야 함
      |             ^~~~
err.c:177:14: error: declaration of ‘arr2’ as array of voids
  177 |         void arr2[7]; // case 83 - void array는 만들 수 없음
      |              ^~~~
err.c:184:18: error: variable or field ‘v’ declared void
  184 |             void v;
      |                  ^
```

전반적으로 과제 프로그램과 거의 비슷하게 출력하였으나, 과제에서 error로 처리한 것을 gcc에서는 warning으로 넘기는 경우도 있었고, error의 원인을 다르게 출력하는 경우도 있었다.